

2章 推薦システムのプロジェクト

2023年1月12日 16:59

2.1 推薦システム開発に必要な3つのスキル

2.1 推薦システム開発に必要な3つのスキル

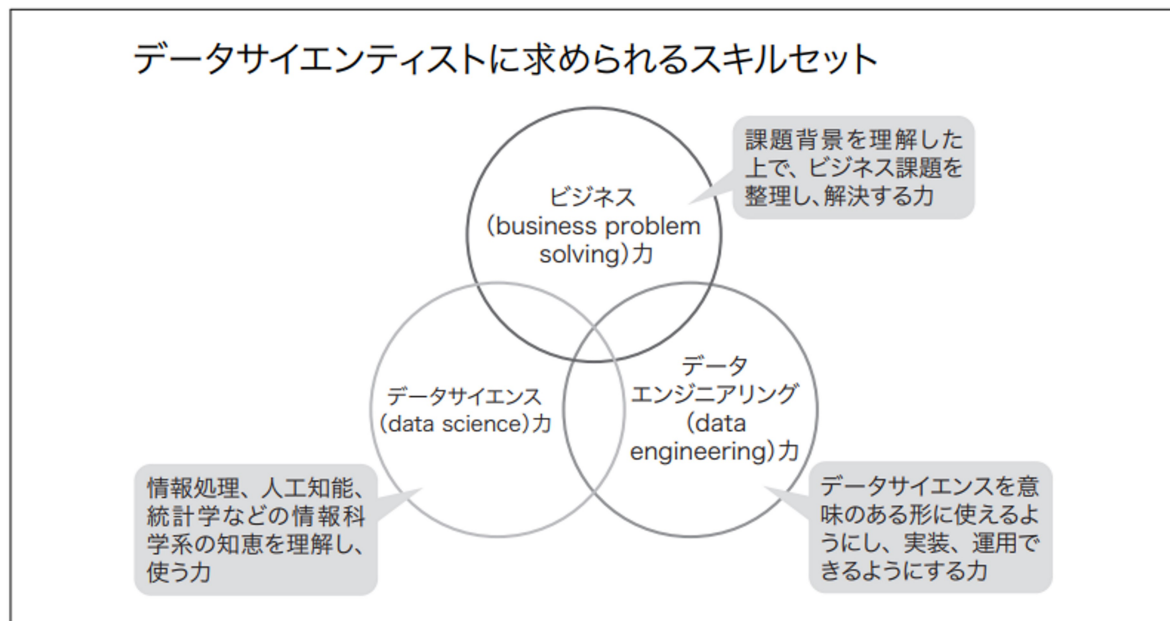


図2-1 データサイエンティスト協会が公開しているデータサイエンティストに求められるスキルセット（参考：2021 年度スキル定義委員会活動報告 2021 年度版スキルチェック＆タスクリスト 公開、https://www.datascientist.or.jp/symp/2021/pdf/20211116_1400-1600_skill.pdf）

- 一般社団法人データサイエンティスト協会（<https://www.datascientist.or.jp/>）
- データサイエンティストに求められるスキルセット
 - 「ビジネス力」
 - 「データサイエンス力」
 - 「データエンジニアリング力」
- ビジネス力
 - ビジネス観点から推薦システムを導入することで何を期待したいのかを定義すること
 - 場合によっては、推薦システムがいらなかったり、人気度順の推薦だけで事足りたりする
 - 推薦システムを導入することで、ユーザーのどの行動の変容を期待するのかといった Key Goal Indicator (KGI) や Key Performance Indicator (KPI) を策定
 - クリック率なのか、購入率なのか、回遊率なのか、Twitterへの投稿数なのか、どのユーザーのどの行動を最大化したいのかによって、作成する推薦システムも異なる
 - KPIの策定には、対象のサービスについて熟知し、サービス上でのユーザーの行動について深く理解している必要がある
 - たとえばYouTube では、ユーザーのトータル視聴時間を KPI として、それが最大化するようにサービス設計がされている。視聴時間が KPI なので、ユーザー が最後まで視聴してくれる動画を推薦することが重要
 - 当初、YouTube では視聴動画数をKPIとして進めていたが、サムネイル画像が目を引く動画ばかりが推薦されるという問題がおきた。そのため、視聴時間を KPI にするように 変更した。
 - その際には、各動画に対して視聴時間のログが細かい粒度で取れていなかったため、視聴時間の正確なログが取れるようにログ取得の整備もした。
 - KPIを策定する際には、現在取れているデータから考えるのではなく、ユーザーの行動の本質を捉えた指標から検討し、もしそのデータが無い場合は、エンジニアと協力しながら、ログの整備をしていくこと
- データサイエンス力
 - ビジネス目標を達成するための理想的な推薦システムを実際に実現可能な推薦システムに落とし込む。

- 求められるデータサイエンス力
 - 扱うデータの量や種類、ビジネスにおける推薦システムの実装の複雑度に応じて異なる
 - いきなり理想とする推薦システムを作ろうとせずに、まずは、王道の手法を使って、どのくらいの精度が出るかを確認して、理想と現実のギャップが大きすぎないかを検討し、適宜ビジネスサイドとコミュニケーションすること
 - その試行 錯誤の過程では、データの性質や各推薦アルゴリズムの長所・短所を把握しておくこと
 - このような俯瞰的なデータサイエンスの知見があることで、試行錯誤の時間を短縮することができる
 - 具体的なスキルと
 - Python や SQL などを利用してデータを加工し、アルゴリズムを構築すること
 - 人気商品や新着商品を推薦する単純な推薦システムの場合は、SQLだけでも構築可能
 - 協調フィルタリングやコンテンツベースの推薦アルゴリズム
 - ◆ 一般に利用できる形のオープンソースソフトウェアとして公開されていることも多く、それらを利用して作ることも可能
 - ◆ 汎用的な推薦アルゴリズムでは十分にアイテムやユーザー情報を考慮できない場合は、独自に 定式化した推薦アルゴリズムを検討し、それをコードに落とし込んでいくことが求められる
 - ◆ その際には、推薦アルゴリズムに関する論文を調査したり、それらを自社 のデータに適用できるように数式やコードを改良したりと高度なデータサイエンス力 が必要
- データエンジニアリング力
 - 実際に推薦システムを実サービスに組み込む際には、さまざまなビジネス要件を考慮した上で実装を行い、システムを安定的に稼働させる必要がある
 - たとえば、1日1回メールでユーザーにおすすめの商品を推薦したいという場合を考えると、全 ユーザーに対する推薦リスト作成の演算が 24 時間以内に終了している必要がある
 - 素晴らしい推薦モデルが定式化されても、計算に100時間かかるとしたら、実ビ ジネスで使うことはできない
 - そのため、計算が早く終わるように処理を並列化
 - データベースの設計やチューニングを工夫
 - 毎日推薦システムに使うデータを加工し、推薦リスト作成演算を行い、推薦リストを データベースに格納するといった一連の処理のパイプラインを整備する
 - これらを安定して稼働させるためには、高度なデータエンジニアリング力が求め られる
- 推薦システム構築において必要となる 3 つのスキルは上記の通り
- これらの 3 つのスキルをすべて兼ね備えた人は稀
- 多くの企業では、それぞれのスキル を持った人々が密にコミュニケーションしながら推薦システムを構築している
- 大企業では、ビジネス部門、データサイエンスチーム、データ基盤チームといった3つ のスキルごとにチームや部署が分かれていることがある
- スタートアップ企業のような社員が少ないところでは、ソフトウェアエンジニアがすべてを担当して、まず は単純な推薦システムを作ることもある

2.2 推薦システムのプロジェクトの進め方

- 3つのスキルを持った人やチームがどのように連携しながらプロジェクトを進めるか
- 7つのプロセス（ECサイト改善の具体例）
 - 1. 課題定義
 - 2. 仮説立案
 - 3. データ設計・収集・加工
 - 4. アルゴリズム選定
 - 5. 学習・パラメータチューニング
 - 6. システム実装
 - 7. 評価、改善
- 1、2、7をビジネス側
- 3、4、5、6、7をデータサイエンティストやデータ エンジニア
- スタートアップ企業など人員が少ない場合は、1人のエンジニアがすべてを担当する場合もある

2.2.1 課題定義

- 自サイトに推薦システムを導入したい場合は、まずは「売上を2倍にしたい」や「会員登録を2倍にしたい」といったビジネス上のそもそもの目的を明確化することが重要
 - たとえば、1ヶ月の売上を2倍にしたい場合には、さらにそれをブレイクダ ウンして、1ユーザーあたりの売上を2倍にするのか、ユーザー数を2倍にするのかなどのビジネス上の指標（KPI）
- KPIの目標と現状の差を確認した上で、現状の自サイトの課題を整理して、適切な打ち手を検討
 - たとえば、AmazonのようなECサイトにおいて1ヶ月の売上を2倍にするために、1人あたりの売上を KPI としてそれを 2 倍にするとい

- 目標を定めたとする
 - 1人あたりの売上を2倍にするにはどこに伸びしろがあるかを分析
 - データを用いて分析する定量的な方法
 - 定量的な分析では、ウェブサイトのユーザーの行動ログのデータを分析することで、「そもそも検索をするユーザーが少ない」や「検索するユーザーは多いが最初の検索結果ページで離脱してしまう」などの現状の課題を知ることができる
 - ユーザーにヒアリングする定性的な方法
 - 「なぜそもそも検索をするユーザーが少ないのか」や「なぜ最初の検索結果ページで離脱してしまうか」といった理由については、データから調査するより、直接ユーザーにヒアリングすることで示唆を得られることが多い
 - ユーザーのヒアリングでは、「どのような単語で検索すれば良いか分からず途中でやめてしまった」や「検索結果の画面で各アイテムのタイトルしか表示されておらず、1つ1つアイテムをクリックして詳細を知るのが大変だった」などの行動ログから読み解くことのできない潜在的なユーザーの心理状態に関するインサイトを得ることができる
 - データとユーザーからのヒアリングの両方のアプローチで、現状の課題の解像度を高めて、今回のビジネスゴールを達成する上での、各課題の重要度を決めていきます。ここでは、重要度が表2-1のように決まったとする

表2-1 課題と重要度

課題	重要度
どのような単語で検索したらよいか分からない	2
検索結果画面の各アイテムの情報が不十分	3
検索結果画面でアイテムが大量にありすぎるためお目当てのアイテムにたどり着けない	4
各アイテムの詳細画面で、そのアイテムに類似したアイテムを知ることができない	4

- データによる分析やユーザーのヒアリング結果として、2つの課題が重要であると判明したとする
 - アイテムが大量にありすぎるために、現状の単純な検索エンジンではお目当てのアイテムにたどり着けず離脱率が高い
 - 各アイテムの詳細画面で、そのアイテムに類似したアイテムを現状知ることができず、そこでの離脱率も高い

2.2.2 仮説立案

- 次に各課題を解決する方法、それを実現するためのコストを検討して、費用対効果（ROI：Return On Investment）が高い施策から取り組む

表2-2 各対応策の優先度

課題	対応策	重要度	コスト	優先度
どのような単語で検索したらよいか分からない	単語のサジェスト機能を実装する	2	中	低
どのような単語で検索したらよいか分からない	検索単語の例をいくつか表示しておく	2	小	中
検索結果画面の各アイテムの情報が不十分	検索結果画面の各アイテムの情報を充実させる	3	小	中
検索結果画面でアイテムが大量にありすぎるためお目当てのアイテムにたどり着けない	検索結果をお気に入り順などでソートできるようにする	4	小	高
各アイテムの詳細画面で、そのアイテムに類似したアイテムを知ることができない	各アイテムの詳細画面に、類似したアイテムを推薦する機能をつける	4	中～大	中

- 今回の例では、検索結果にアイテムが大量にありお目当てのアイテムにたどり着けないという課題に対しては、そこに人気度順や価格順でソートできる機能を追加するだけで、ユーザーがお目当てのアイテムを探しやすくなり、離脱率を下げるができる
 - このケースでは、まずは検索周りの実装コストが低く重要度が高い課題を解決してから、推薦システムに取り掛かるのが効果的
- このように、わざわざ推薦システムを導入するより、他にビジネスゴールを解決できるコストが低い方法がある場合はそちらをまず検討するのが大切
- 特に、検索システムの機能改善をしたほうが費用対効果が高い場合も多いため、わざわざ推薦システムを導入せずに、検索機能を拡張する方向性も検討する
- 実際のビジネス場面では、上からの号令で、「推薦システムを導入するぞ」と推薦システムの導入が決定事項で下りてくることもあるかもしれない。そのときには、一度立ち止まって、本当に、現状の課題に対して推薦システムの導入が一番費用対効果が高いのかを再検討することが大切
- 著者自身も、推薦システムを導入したいという相談を何度も受けたことがあるが、話をよくよく聞いてみると、検索システムを改善したほうが良かったり、デザインを変えたほうが良かったりするケースも多くあり、実際にそちらを改善することで、クリック率やコンバージョン率も増加した。
- 検索周りの改修が完了し、それらの課題が解決されたとする。次に重要な課題である各アイテムの詳細画面での類似アイテムの推薦機能に着手する。コストのところで、中～大としているのは、ひとくくりに類似度推薦と言ってもいくつかのやり方があるため
- 今回のビジネスフェーズにおいて適切な推薦アルゴリズムを選ぶ必要がある。どのような類似度推薦を実装するかは、ユーザーのニーズをヒアリングして整理していく
 - 手芸品の販売サイトでは、とある作品に対して同じ作者の作品を類似作品として並べるだけでも効果があるかもしれないし、作品の

ジャンルが同じものを並べるほうが効果があるかもしれない

- 音楽のストリーミングサイトでは、音楽の曲調が似ている曲より、その曲が流行った時期の別の曲を推薦することが効果的かもしれない
- Amazonのように「この商品を買った人はこんな商品を買っています」という類似度推薦が効果的かもしれません。このようにアイテムの類似度と言っても、いくつかの観点があるので、ユーザーが一番求めている類似度は何なのかを把握した上で実装していくことが大切

2.2.3 データ設計・収集・加工

- ・ 現状どのようなデータが蓄積されているかを整理する
- ・ 推薦システムの開発に必要なデータは主に次の2種類がある
 - ユーザー・アイテムのコンテンツ情報
 - ユーザーの情報は、年齢や性別などのプロフィール情報や、どのジャンルのアイテムを好むかといった嗜好情報
 - アイテムの情報は、アイテムの説明文やタグ、カテゴリ、登録日などの情報
 - ユーザーとアイテムのインタラクション情報
 - 閲覧やブックマーク、購入、視聴、評価の情報
- ・ まず自社サービスがどのデータを保有しているかを整理して、そのデータで推薦システムが構築できるかを検討する
 - 特にサービスの初期段階では、ユーザーとアイテムのインタラクションのデータがないことも多く、まずは、アイテムの情報を利用した推薦システム構築から検討することが多い
- ・ たとえば、アイテムに「食品」や「電子機器」といったカテゴリを表すタグが付与されている場合は、アイテムのタグデータを使って、タグが一致したものを推薦するという推薦システムを実装していくことを考える
 - まずは、そもそも各アイテムにちゃんとタグが付いているかの確認が必要
 - もし、一部のアイテムにタグが付いていなかったり、間違っていていたりするものが多くある場合は、人手でタグの整備をする必要がある
 - それが大量にある場合は、クラウドソーシングなども利用して推薦システムに使える形に整える
- ・ また、ユーザーの閲覧ログを使って、「このアイテムを閲覧している人はこのアイテムを閲覧しています」という推薦システムを実装する場合には、閲覧ログからボットなどによる不適切なログを取り除いたデータを準備する
- ・ このように、必要なデータが適切に存在するかを確かめた上で、推薦システムに使えるように前処理をする必要がある

2.2.4 アルゴリズム選定

- ・ どの推薦アルゴリズムを使用するかを決めていく
- ・ アルゴリズムの計算時間、必要なデータ、求められる予測精度などのさまざまな観点から、今回のビジネスゴールに適するものを選ぶ
- ・ 図2-2のように、一般の機械学習モデルと同様に一定のコストをかけると最初はある程度の精度が出る
- ・ しかし、精度が95%のものを96%にするには多大なコストがかかる場合がある
- ・ サービスの初期においては、まずは簡単なアルゴリズムから実装していくことが大切

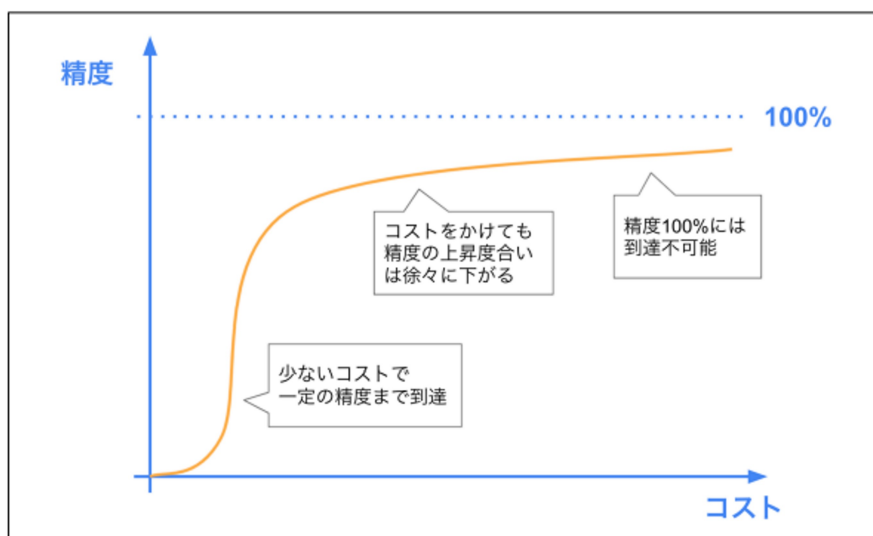


図2-2 費用対効果の関係

2.2.5 学習・パラメータチューニング

- ・ 推薦システムの学習とチューニングを行ってリリースする前に、過去のデータを利用してオフラインで推薦システムの良さを検証する
- ・ オフラインで推薦システムの精度を検証する場合
 - RecallやPrecisionといった機械学習の指標も重要

- しかし、具体的にこのアイテムを入れるとどのようなアイテムが出てくるのかを確認して、それに納得度があるかを確認することも重要
 - 特にエンジニアがビジネスサイドと議論するときには、推薦結果の具体例を提示しながら話すとイメージが湧きコミュニケーションが円滑になる
 - 古いアルゴリズムでおすすめされるアイテム群はこれで、新しいほうだとこのように変わりますと具体例を交えながら話すと、アルゴリズムの差分が分かりやすい
- オフラインで検証する際には、データのバイアスに注意する必要がある
 - Amazonの5つ星の評価のデータには、各星の評価値が均等に存在するのではなく、5つ星が多いというバイアスがある
 - 商品に対して特別な思いがある人が評価をつけやすいという傾向があるため
 - 検索エンジンの影響を受けるバイアスもある
 - 検索の上位に現れるものは、クリックされやすく、下位にくるものはクリックされづらい
 - このデータをもとに、なんの処理もせずに行動履歴ベースの推薦システムを作ると、検索エンジンの影響を受け、検索上位のものがより推薦されやすくなったりする
 - そのため、これらのバイアスを取り除いて、推薦システムを作り、評価することが大切

2.2.6 システム実装

- オフラインで良い推薦アルゴリズムが完成したら、いよいよ実システムに組み込む
 - 推薦アルゴリズムの学習や予測の更新頻度
 - 新規のアイテムやユーザー に対しての推薦
 - 推薦に関連するデータのパイプラインの設計
 - などを考慮
- 実ビジネスでよく利用されているバッチ推薦とリアルタイム推薦の2種類について概要（詳しくは6章で）
 - バッチ推薦
 - 1日1回や1週間に1回など決まったタイミングでその時点の情報をもとに推薦リストを更新し、ユーザーに対して提供
 - 推薦リスト作成に必要な処理は2ステップ
 - 推薦モデルの学習
 - ◆ 毎回学習し直したほうが新規アイテムに対しても推薦が可能になるので良いが、学習に時間がかかる場合は、妥協して1週間に1度の学習にすることもある
 - 推薦モデルによる予測
 - ◆ ユーザーが何人いるかによって計算の仕方が変わる
 - ◆ ユーザーが10万人いて1人1人に推薦リストを作る場合
 - ◇ 1人あたりの推薦リスト作成時間を1秒とすると、愚直にやると10万秒=27.8時間かかる
 - ◇ これを高速化するためには、並列処理をしたり、1人あたりの推薦リスト作成時間を減少させる必要がある
 - リアルタイム推薦
 - ユーザーの直近の行動履歴を即時に反映した推薦リストを作るもの
 - ユーザーの行動履歴を即時に反映させるには幅広いエンジニアリングの技術が必要
 - データの同期
 - リアルタイム演算
 - など
- このように実システムに組み込む際には、ユーザー数やアイテム数の量によって、また推薦のリアルタイム性によってもシステム構成が変わる

2.2.7 評価、改善

- 最後に、推薦機能をリリースして、それが実際に効果があったかを検証する
- A/Bテストのような仕組みで、一部のユーザーのみに推薦機能を提供して、推薦ありの場合となしの場合におけるユーザーの購入金額や滞在時間を検証する
- A/Bテストが難しい場合は、リリース前後での比較になるが時系列のトレンドなどを考慮した上で評価することが必要
- また、推薦システムを導入したことで推薦経由の売上が発生したとしても、それを成功とみなすのは尚早
- 推薦経由での売上が上がっても、検索経由の売上が下がり、トータルで1ユーザーあたりの売上が下がっていることもある
- 推薦システム単体を見るのではなく、全体を見て悪影響がないかを確認するのも大切
- 著者自身も、推薦システムを導入した直後にログを確認して、推薦システム経由のコンバージョンが増加して嬉しく思っていたところに、隣の検索チームから、検索の利用が激減しているという一報を受けた。
 - 調べてみると、推薦システムによる回遊が増え、検索の利用が減っており、全体的にはコンバージョンは増加しているという結果だった
- 推薦システムを開発するときは、関係するチームに事前に機能の内容、影響範囲、リリース日などを相談しておくことが大切
- リリース後のユーザー行動を分析することで当初の仮説を検証することができる

- 想定よりクリックや売上が伸びない場合には、ユーザーの行動ログ分析やユーザーインタビューで原因を調査する
- 原因の候補に当たりがつけば、それを改善して、再度リリースするプロセスを繰り返す

2.3 まとめ

- 推薦システムのプロジェクトを進めるのに必要な3つのスキル（ビジネス力、データサイエンス力、データエンジニアリング力）と、プロジェクトの進め方を7つのプロセスで解説した
- プロジェクトを進める上で、ビジネスゴールを定めて、現状の課題の解像度をデータとユーザー調査から高め、費用対効果が高い施策を行うことが特に大切
- その際には、推薦システムを使わない施策も検討することが重要