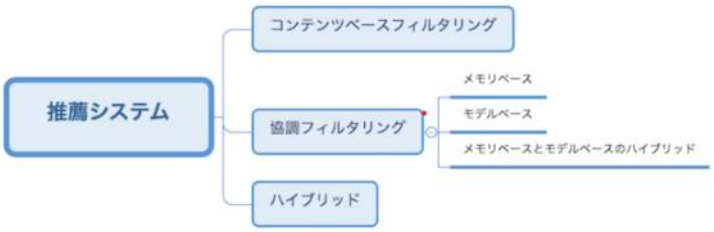
11/12 調べもの

2022年11月12日　14:02

参考サイト
協調性フィルタリング
コンテンツベース
数理的解説
レコメンド研究のあれこれ
recbole
全体像



コンテンツベース(アイテムの特徴をもとにユーザが過去に高評価したアイテムを類似したアイテムをレコメンド)
- 他ユーザのデータが不要なため、ユーザにパーソナライズしたアイテムをレコメンドできる
- スケールが楽(アイテムの特徴と対象ユーザの情報だけでレコメンドできるため)
- ユーザの嗜好をとらえるためニッチなアイテムもレコメンドできる

- ｓアイテムの特徴表現するのでドメイン知識が必要になる(要するに手作業)
- レコメンドされるアイテムがユーザの既存の嗜好に制限される(セレンディピティに乏しい)

メモリベース(メモリにロードされたデータセットに基づいたレコメンド)
- 容易に実装可能、新規データを容易に追加できる
- アイテムの特性を把握する必要がない
- 共通に評価されているアイテムについてはスケールする

- 人の評価に依存する
- データが疎な時にパフォーマンスが低下する
- 新規ユーザやアイテムをレコメンドできない
- 大量データのスケーラビリティに限界がある

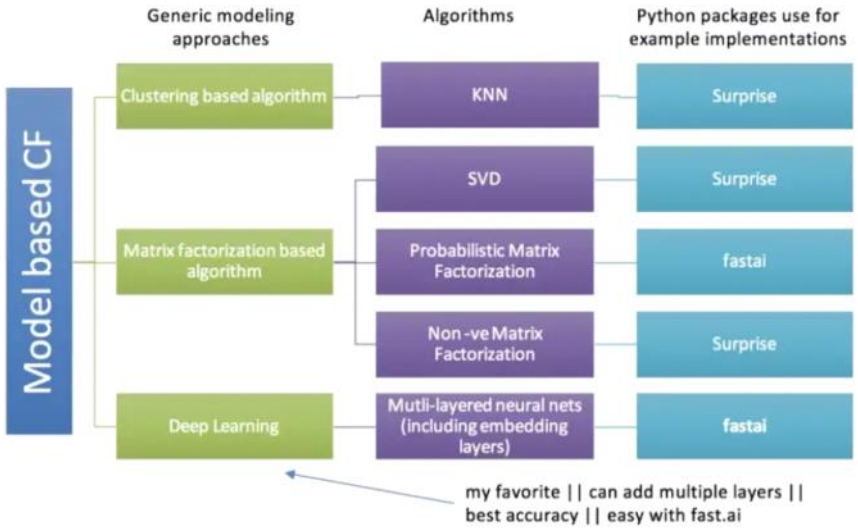
ユーザベース協調フィルタリング(user base collaborative filtering, UBCF)ーユーザ間の類似度に応じてレコメンド
  - 新規ユーザが加えられるたびに類似度を再計算する必要があるためオンラインでのレコメンドはコストがかかる

アイテムベース協調フィルタリング(item base collaborative filtering, IBCF)ーアイテム間の類似度に基づいてレコメンド
  - 新規ユーザが追加されてもアイテム間の類似度を再計算する必要はない。コスト削減できる(オフラインで計算すればよい)

一般的な情報
ユーザベースよりもアイテムベースのほうが好まれる
→扱うアイテム数がユーザよりもアイテムのほうが圧倒的に多いため
　ユーザがアクションするアイテムがアイテム全体のうちわずかであるため

モデルベース(データセットから情報を抽出しレコメンド→機械学習もこれに相当する)
- ロバスト性と精度向上のために特異値分解(SVD)や主成分分解といった次元削減がよく用いられる
- 疎な高次元配列ではなく低次元の非常に小さいサイズの行列を扱える
- 次元圧縮を行うため大規模なデータセットを用いる場合は非常にスケーラブル(次元圧縮によって得られた行列の類似度を比較すればよいため)
- 予測精度が向上する
- レコメンドが直感的な根拠に基づく(説明可能)

- モデル構築コストが高い
- 予測精度とスケーラビリティはトレードオフ
- 次元削除によって情報が損失

- 

    - 行列分解について
        - 一般的にメモリベース型の協調フィルタリングに比べて、実装は複雑であるが推薦の性能は良い
        - ユーザとアイテムを100次元ほどの低次元ベクトルで表現し、その内積値を相性としている
        - それぞれの手法の違いの観点としては①欠損値の取り扱い②評価値が明示的か暗黙的か(明示的=顧客に評価を質問して答えてもらう、暗黙的=顧客の行動に基づいて推定する)
        - movielensは明示的なデータ
        - データから自動的に軸を決定し次元を圧縮する
    - 大まかな分類
        - SVD(特異値分解)
            - 欠損個所に0または平均値を代入し、特異値分解を行う
            - 平均値のほうが性能は良い
            - ただ、0の方が相対値には意味がある
        - NMF(非負値行列分解)
            - 行列分解をしたときに要素が0以上となるように制約を入れたもの
            - 欠損部分には0を入れるため性能は低い
        - MF(Matrix Factrization、明示的な評価値に対する行列分解)
            - 大規模なデータに対してはSVDやNMFは避けた方がよい
            - SVDのように欠損値を穴埋めすることはなく、観測された評価値のみを使って行列分解する
            - SparkやBigQueryなどでも実装されている
        - IMF(暗黙的な評価値に対する行列分解)
            - 顧客の行動から評価値を算出する
                - 例えば商品を一回でもクリックすれば1、なければ0
                - 明示的に比べてデータがとりやすい
                - ただし、ノイズも多い
                - 行動回数(クリック回数)も目的関数に入れたうえで行列分解
        - BPR(Bayesian Personalized Ranking)
            - 暗黙的な評価値を用いている
            - ユーザごとの嗜好をランキングとして学習する
            - 全てのデータからサンプリングして学習
            - 計算コストが魅力的
            - 並列化は難しい
        - FM(Factorization Machines)
            - ユーザやアイテムの属性情報を使って推薦システムの性能を上げる
                - 新規のアイテムやユーザに対して推薦ができないコールドスタート問題にも対応できる
            - 入力データの形式が異なる
            - 1つの評価に対する情報が1行で表される(評価数×特徴量数)
            - 特徴量同士の組み合わせも考慮することができる
            - 多項式モデルであるが重みは線形項にしかない設計