日本データベース学会和文論文誌
Vol. 18-J, Article No. 3, 2020 年 3 月

推薦システムにおける推薦理由提示手法の提案 ー機械学習解釈モデルを用いてー

森澤 竣♡ 真鍋 智紀♧ 座間味 卓臣♢
山名 早人♤

推薦システムにおいて，アイテムの推薦理由をユーザに提示することは，推薦の効果や透明性，そしてユーザの満足度を向上させることが示されている．推薦理由の説明をするためには，推薦モデルの解釈が必要となる．しかし，機械学習を用いた近年の推薦モデルは解釈が難しいブラックボックスとなっているものが多く，学習済みモデルの状態から推薦理由を提示することは困難となっている．そこで本稿では，機械学習モデルへの入力とその出力との関係を解釈する方法である LIME を推薦システムに適用することを提案する．具体的には，任意の学習済み推薦モデルが与えられたとき，当該推薦モデルに対する入力—出力のペアを線形回帰モデルで学習させ，その結果得られた各特徴の重みを重要度として推薦モデルを解釈する．提案手法はあらゆる推薦モデルに対して適用できる．また，提案手法は推薦システム自体に変更を加えないため，推薦の精度に影響を与えない．評価実験では，解釈性の高い協調フィルタリングを使用したレーティング予測モデルに提案手法を適用することによって，LIME を使用した特徴量の重要度の算出が正確であることを検証した．LIME の出力による重要度の高い特徴量と，協調フィルタリングモデルで実際に出力の算出に影響した特徴量の適合率による評価から，提案手法は重要度の高い特徴量が特定可能であることを示した．

1. はじめに

説明可能な推薦システムとは，アイテムの推薦に加えて，推薦理由をユーザに説明することができるシステムを意味する．推薦システムのアルゴリズムの一つとして有名な協調フィルタリングは解釈性が高く，説明を提供しているサービスが多く存在する．協調フィルタリングを使用した推薦システムでは，例えば，「あなたに似たユーザはこのような商品を購入しています」「あなたが購入した商品に関連したこのような商品を推薦します」といった説明をすることが可能である．こうした推薦理由の説明は，推薦の効果や透明性，そしてユーザの満足度を向上させることが示されている[1].

近年は，ニューラルネットワークをはじめとした，機械学習アルゴリズムを用いた新たな推薦モデルが多く提案されている．機械学習アルゴリズムを使用した推薦システムは協調フィルタリングよりも高い精度を発揮するが，解釈性が低いため，ユーザに対して説明を与えることが難しい．そこで，従来ブラックボックスとされてきた機械学習モデルを解釈するための研究や，説明可能な新しい機械学習アルゴリズムの研究が行われている．説明が可能な機械学習アルゴリズムを用いたシステムは，一般的に「説明可能な AI(XAI: Explainable AI)」と呼ばれ，機械学習を活用するあらゆる分野で関心が高まっている[1].

機械学習を用いた説明可能な推薦モデルの研究としては，S. Seo ら[2]のアテンションニューラルネットワークを使って解釈を可能にする手法や，B. Abdollahi ら[3]の説明可能でない学習結果にペナルティをかけることによって最終的な学習結果が説明可能となるように矯正する手法が存在する．しかし，[2][3]の手法は学習結果を解釈可能にするために元の機械学習モデルに対して手を加えており，元のモデルに比べて推薦自体の精度の下がるケースが存在する．

本稿では，機械学習モデルの解釈を行うための手法として M. T. Ribeiro ら[4]によって提案された LIME(Local Interpretable Model-agnostic Explanations)を推薦システムに組み込むことによって説明を生成する新たな手法を提案する．LIME は，任意の機械学習モデルに対する入力を説明変数，出力を目的変数として解釈可能なモデルである線形回帰モデルに学習させ，各特徴の重要度を求めることによって解釈を行う手法である．また，複雑な推論アルゴリズムに対応するために，特徴空間内の予測対象のベクトルの近傍を学習することによって，局所的に正しい説明を生成する．LIME は推論結果の説明として，出力に対する各入力特徴の重要度を数値として捉えることができるので，推薦システムでは，重要度の高い特徴を説明としてユーザに提示することできる．提案手法を用いることによって，あらゆる推薦モデルに対して，推薦の精度を落とさずに解釈をすることが可能となる．

本稿では，以下の構成を取る．第 2 節では説明可能な推薦システムの関連研究について述べる．第 3 節で提案手法について説明し，第 4 節で評価実験の結果，および結果の考察について述べる．最後に，第 5 節でまとめを行う．

2. 関連研究

本節では，近年の機械学習を使用した推薦モデルに対して，モデルを解釈してユーザに説明するための研究について，推薦モデルのタイプ別に分類して説明する．

2.1 行列分解を用いた説明可能な推薦モデル

本項では，行列分解(Matrix Factorization)を用いた説明可能な推薦モデルに関する研究について紹介する．行列分解による分解後の各要素は，ユーザもしくはアイテムを表す潜在的な因子として仮定されているが，それぞれの要素がどのような意味を持っているかを解釈することは難しい．

Y. Zhang ら[5]は，テキストによるユーザレビューから単語レベルで製品の特徴を抽出し，行列分解における各潜在的な次元

♡ 学生会員 早稲田大学大学院基幹理工学研究科
hiroshun@yama.info.waseda.ac.jp
♧ 非会員 早稲田大学大学院基幹理工学研究科
tomoki_manabe@yama.info.waseda.ac.jp
♢ 非会員 合同会社 DMM.com
zamami-takumi@dmm.com
♤ 正会員 早稲田大学理工学術院
yamana@yama.info.waseda.ac.jp

[1] 高野敦(2018)「もうブラックボックスじゃない, 根拠を示して AI の用途拡大」, 『日経エレクトロニクス』2018 年 9 月号, pp. 53-58, 日経 BP 社.

をユーザレビューから抽出した特徴に対応させるモデルを提案した．特徴量が各単語に対応しているので，単語を用いてユーザに対して説明を行うことが可能である．本手法を用いた説明を加えた推薦は，説明を加えなかった場合に比べて 1.14%の CTR (Click Through Rate)を向上させるという結果がオンライン実験によって得られている．本手法の問題点は，レビューの件数が少ないアイテムは推薦されにくい点である．

B. Abdollahi ら[3]は，行列分解を用いたレーティング予測モデルにおいて，「あなたに似たユーザはこの商品を高く評価しました.」といった説明の提示を前提とした，説明可能な行列分解モデルに関する研究を行った. ユーザ-アイテムの組み合わせに応じて，上記の説明を与えたときの説明の正確さを表す指標である Explainability(説明可能性)を，説明提示対象のユーザの過去のレーティングに対する予測レーティングの期待値を用いて定義した．さらに，行列分解の最適化における目的関数に Explainability を加えることによって，説明が正しくなるアイテムのみが推薦されるように改良を行った．また，[6]では，同様に目的関数に Explainability を組み込む手法を，制限付きボルツマンマシン(Restricted Boltzmann Machines)を用いる推薦モデルに適用する手法を提案した. [3],[6]の問題点は，元の推薦モデルの目的関数に Explainability を組み込まない場合と比べて，モデルのパラメータによっては推薦の精度が下がるという点である．

2.2 グラフ構造を用いた説明可能な推薦モデル

本項では，グラフ構造を用いた説明可能な推薦モデルに関する研究について紹介する．グラフベースの推薦モデルはソーシャルネットワーク関連の推薦モデルにおいてユーザ間あるいはユーザ・アイテム間の関係をグラフとして表すことによって活用されている．

X. He ら[7]は，ユーザをモデル化するためにユーザとアイテム，そしてアイテムの特性を表すアスペクトの関係を表す三部グラフを推薦モデルに導入した．推薦モデルは，三部グラフがユーザの興味とアイテムの特性を表すように最適化を行い，アイテムを表すノードをランク付けすることによってアイテムを推薦する．アスペクトのノードはユーザレビューから取得した単語レベルのアイテムの特性であるため，グラフ構造を読み取ることによってユーザの好みのアスペクトを提示することが可能である．本手法によるトップ N 推薦は，アイテムベースの協調フィルタリングを使用したトップ N 推薦結果と比較して, HR(Hit Ratio)を 1.13, nDCG(normalized Discounted Cumulative Gain)を 0.8 向上させる結果が得られているが，レビューの少ないユーザに対しては，推薦の精度が下がるという問題点がある.

R. Heckel ら[8]は, 購入履歴等のユーザとアイテムの関係を表す二部グラフに基づいた，重複ありのクラスタリングによる説明可能な推薦手法を提案した．各クラスタに含まれるユーザは同様の興味があると解釈することができるので，「アイテム A を購入したあなたと同様の興味を持つユーザ X はこれらの商品を購入しました」といった説明が可能である．データセットにおけるユーザのアイテムへのレーティングの有無を正解ラベルとして評価し，ユーザベースの協調フィルタリングに比較して MAP(Mean Average Precision)値が 0.027 向上したと報告している．本手法の問題点は，各クラスタの具体的な特性を説明することが難しい点である．

2.3 ニューラルネットワークを用いた説明可能な推薦モデル

本項では，ニューラルネットワークを用いた説明可能な推薦モデルに関する研究について S.Seo らの研究[2]と C. Chen らの研究[9]を紹介する．いずれの研究も，入力の特徴の重みを解釈しやすくするためにアテンションニューラルネットワークを使用している．アテンションニューラルネットワークとは，入力ベクトルの各要素に重みを与える層(Attention Layer)を追加したニューラルネットワークである．

S. Seo ら[2]は, 入力にユーザが過去に書いたレビュー文章と，アイテムが過去に書かれたレビュー文章の単語を Bag of Words によって表現したベクトルを入力として，ユーザがアイテムに与えたレーティングを予測するアテンションニューラルネットワークを提案した．Attention Layer における数値を，単語埋め込みベクトルに対応する各単語の重要度として解釈することができるので，数値の高い単語はユーザの興味のある特徴，もしくはアイテムに関係する特徴として，ユーザに説明することが可能である．本手法を用いた推薦は，レーティング予測における, 最小二乗誤差が行列分解モデルと比較して 0.08 減少させる結果が得られているが，レビューの少ないユーザに対する推薦の精度は低いという問題点がある．

C. Chen ら[9]は，レビュー文章を一度畳み込みニューラルネットワークによってベクトル化し，これらのベクトルを入力として，ユーザがアイテムに与えたレーティングを予測するアテンションニューラルネットワークを提案した. Attention Layer における数値は，各レビュー文章の重要度として捉えることができるため，過去に書かれたレビュー文章をユーザに対する説明として提示することが可能である．本手法を用いた推薦は，レーティング予測における，最小二乗誤差を減少させる結果が得られている．一方で，過去に書かれたレビュー文章をユーザに提示することは，商品の欠点が記されているものなど，推薦として妥当ではないレビュー文章が提示される可能性があるという問題点がある．

2.4 関連研究のまとめ

本節では，近年の機械学習モデルを使用した説明可能な推薦システムにおける既存研究を紹介した．既存研究の問題点の 1 つとして，既存モデルを解釈するために手を加えることが，推薦モデルの精度を下げることがある点が挙げられる．例えば，[3]では, 目的変数に Explainability に基づいたペナルティを加えることによって，単なる行列分解よりも精度を下げるケースが

存在している．また，既存研究では，それぞれの推薦システムや機械学習モデルに適した説明のための手法が提案されているが，様々なモデルに簡単に組み込めるようにするために，あらゆるモデルに対して適用可能な手法が求められる．

3. LIME を使用した推薦理由提示手法の提案

本節では，提案手法である LIME を使用した推薦理由の提示手法について述べる．3.1 項では，機械学習解釈モデルである LIME のアルゴリズムについての説明を行う．3.2 項では提案手法について説明を行う．

3.1 LIME のアルゴリズム

LIME(Local Interpretable Model-agnostic Explanations)[4]は，任意の学習済み機械学習モデルに対して，推論の結果を説明するためのアルゴリズムである．具体的には，入力ベクトルの近傍の領域に関して，任意の入力ベクトルを生成し，その入力をもとに学習済み機械学習モデルから出力を得る．そして，この入力と出力のペアを用いて別に用意した線形回帰モデルを学習させる．この結果得られた線形回帰モデルから，元の学習済機械学習モデルの推薦結果を説明する手法が LIME である．LIME では説明として，推論に影響を与えた特徴量の重要度を出力する．本手法は，特徴を表現した入力に対して出力を与えるあらゆるモデルに対して適用が可能である．

[4]では，分類問題を推論する機械学習モデルに対して，LIME を適用して説明を生成するための手法が提案されている．さらに，[4]の筆者による実装[2]では，回帰モデルへ適用可能なように拡張されている．回帰モデルにも適用可能な LIME による説明生成のアルゴリズムを，アルゴリズム 1 に示す．

アルゴリズム 1　LIME を用いた説明の生成[4]

入力: 説明対象のモデルf，入力ベクトル$\boldsymbol{x}$
入力: サンプル数N，類似度カーネル関数π_x
入力: 説明に用いる特徴量の数K
出力: 各特徴量の重要度$\boldsymbol{w}$

1. $\mathcal{Z} \leftarrow \{\}$
2. **for** $i \in \{1, 2, 3, \ldots, N\}$ **do**
3. $\quad z_i \leftarrow \text{sample\_around}(\boldsymbol{x})$
4. $\quad \mathcal{Z} \leftarrow \mathcal{Z} \cup \langle \boldsymbol{z_i}, f(\boldsymbol{z_i}), \pi_x(\boldsymbol{z_i}) \rangle$
5. **end for**
6. $w \leftarrow \text{K} - \text{Lasso}(\mathcal{Z}, K)$
7. **return** w

入力には，説明対象のモデル$f \in \mathbb{R}^d \rightarrow \mathbb{R}$と，入力ベクトル$\boldsymbol{x} \in \mathbb{R}^d$を与える．また，パラメータとして，サンプリングするベクトルの数Nと説明に用いる特徴量の数 K，そして，$\boldsymbol{x}$との類似度を算出する類似度カーネル関数 π_xを与える．π_xは，局所性をコントロールする関数であり，式 3.1 で表される．σはカーネル幅，Dはコサイン距離やユークリッド距離といった距離関数を表す．

$$\pi_x(z) = \exp\left(-\frac{D(x,z)^2}{\sigma^2}\right) \qquad \text{(式 3.1)}$$

まず，推論結果の説明を行う入力ベクトル$\boldsymbol{x}$を中心として，ベクトルをサンプリングする(3 行目). サンプリングした入力ベクトル$\boldsymbol{z_i}$と説明対象のモデルによる出力$f(\boldsymbol{z_i})$，そして入力ベクトル$\boldsymbol{x}$と$\boldsymbol{z_i}$の類似度$\pi_x(\boldsymbol{z_i})$を集合$\mathcal{Z}$に追加する(4 行目). 3，4 行目の操作をN回分行う．そして，z_iを説明変数，$f(\boldsymbol{z_i})$を目的変数として，K個の特徴量のみを用いた複数の線形回帰モデルの学習し，説明に最適なモデルの選定を行う(6 行目). 最適な線形回帰モデルを$\xi(\boldsymbol{x})$，候補の線形回帰モデル$g \in G$とすると，$\xi(\boldsymbol{x})$の選定は式 3.2 に基づいて行われる．$\xi(\boldsymbol{x})$には，$\boldsymbol{x}$の局所性π_xを考慮した，fのgによる近似の損失関数Lを用いる．Lの計算式を式 3.3 に示す．また，関数Ωは，モデルgの説明の複雑度を表し，gの重みのベクトルのうち非ゼロの要素数を返す関数が一般的に用いられる．

$$\xi(\boldsymbol{x}) = \boldsymbol{w}\boldsymbol{x} = \underset{g \in G}{\text{argmin}}\ L(f, g, \pi_x) + \Omega(g) \qquad \text{(式 3.2)}$$

$$L(f, g, \pi_x) = \sum_{z \in Z} \pi_x(\boldsymbol{z})\big(f(\boldsymbol{z}) - g(\boldsymbol{z})\big)^2 \qquad \text{(式 3.3)}$$

以上の方法によって選定された線形回帰モデルξにおける重み$\boldsymbol{w}$が出力となる．

アルゴリズム 1 によって出力された，重み$\boldsymbol{w}$を用いて説明のために可視化した一例を図 3.1 に示す．図 3.1 では，重み$\boldsymbol{w}$の要素のうち，非ゼロであった要素と対応する特徴量の名前を棒グラフによって可視化している．正の重みを持つ特徴は，目的変数を正の値にするために寄与した特徴であり，負の重みを持つ特徴は，目的変数を負の値にするために寄与した特徴であると解釈することができる．

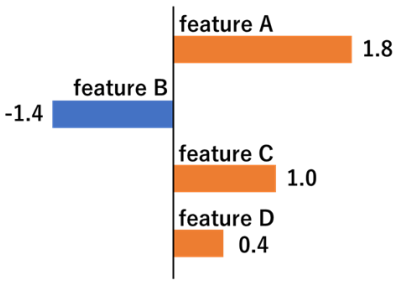

図 3.1 LIME による説明の可視化の一例

3.2 推薦理由提示手法の提案

以下では，任意の推薦モデルに対して LIME を適用し，算出された特徴量の重要度を用いてユーザに推薦理由の提示を行う手法を提案する．図 3.2 に提案手法の模式図を示し，以下に

[2] GitHub - marcotcr/lime: https://github.com/marcotcr/lime

各番号の手続きに対応した説明を記す.

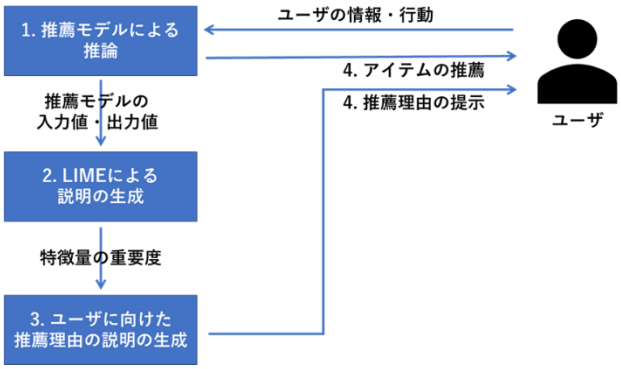

図 3.2 提案手法の模式図

1. ユーザの情報・行動データやアイテムのデータなどから，任意の学習済み推薦モデルを用いて推薦するアイテムを決定する.
2. LIME によって，1 で決定したアイテムの推薦理由を生成する．具体的には，アイテムの推薦時に使用した，アイテム・ユーザを表現する入力ベクトルの持つ各特徴量の重要度を算出する.
3. 各特徴量の重要度を用いて，ユーザに提供する推薦理由の説明インタフェースを生成する．推薦理由の説明には，図 3.1 のように特徴量の重要度を図によって提示することが可能であるが，ユーザの推薦理由への理解を深めるためには，文章として提示する方法が妥当であると考えられる．特徴量の重要度を用いた説明文を生成する方法としては，あらかじめ各特徴量を説明する定型文を用意し，各推薦に対する重要な特徴量に応じて定型文を出し分ける手法が考えられる．例えば，あるアイテムiに対して与えたレーティングを表す特徴量では「あなたはアイテムiを高く評価したため, 以下のアイテムを推薦します.」といった文章，アイテムのジャンルgを表す特徴量では「あなたはジャンルgに興味があるので，以下のアイテムを推薦します.」といった文章を提示することが考えられる.
4. 2 に基づいたアイテムの推薦と同時に，3 で生成した推薦理由をユーザへ提示する．説明の提示は，ユーザの推薦システムに対する透明性，信頼性の向上や，推薦アイテムに対する興味の向上が期待できる.

提案手法は，LIME を用いることによってあらゆる推薦モデルに対して適用が可能である．また，推薦モデル自体に変更を加えないため，推薦の精度に影響を与えない．以上の点は，第 2 節に記述した近年の機械学習を用いた説明可能な推薦モデルにおける問題点を解決する.

4. 評価実験

本節では，提案手法に対する評価実験の手法と実験結果について述べる.

4.1 評価実験の概要

本実験では，提案手法において LIME を適用することが妥当であることを検証するために，LIME による説明の正確性を複数の評価指標によって計測した．具体的には，解釈が容易な推薦モデルである協調フィルタリングに本手法を適用し，協調フィルタリングの内部のアルゴリズムの解釈による特徴量の重要度の数値と，LIME によって生成された特徴量の重要度の数値の比較を行った.

4.2 協調フィルタリングの実装

提案手法の適用を行う推薦モデルとして，B. Sarwar ら[10]によるアイテムベースの協調フィルタリングを実装した．[10]は，過去にユーザがアイテムに与えたレーティングのデータから，特定ユーザが特定アイテムに与えるレーティングの予測を算出する推薦モデルである．現実の推薦システムでは，推薦モデルを用いてユーザが各アイテムに与える予測レーティングを計算し，レーティングの高いトップ N アイテムを推薦する方法で用いられている.

レーティングの予測値は，以下のように算出される．ユーザの集合を$U = \{u_1, u_2, \dots, u_m\}$，アイテムの集合を$I = \{i_1, i_2, \dots, i_n\}$としたとき，ユーザ$u$がアイテム$i$に与えたレーティングを$R_{u,i}$とする$m \times n$の行列$R$を定義する．ここで，アイテム$i$とアイテム$j$の類似度を$s_{i,j}$を，コサイン類似度を用いて式 4.1 のように算出する．なお，コサイン類似度の計算には，Rにおけるi列目とj列目のベクトルのうち，両方ともレーティングを与えているユーザによる要素のみを抽出したベクトル$\boldsymbol{i}, \boldsymbol{j}$を用いる.

$$s_{i,j} = \cos(\boldsymbol{i}, \boldsymbol{j}) = \frac{\boldsymbol{i} \cdot \boldsymbol{j}}{\|\boldsymbol{i}\|_2 * \|\boldsymbol{j}\|_2} \quad \text{(式 4.1)}$$

式 4.1による類似度を用いて，uが過去に評価した類似アイテムのうち，類似度の高い上位Nアイテムを選択する．アイテムの選択数Nはモデルのパラメータとして指定する．類似度の高いNアイテムを用いて，ユーザuがアイテムiに与える予測レーティング$P_{u,i}$の算出を行う(式 4.2).

$$P_{u,i} = \frac{\sum_{j \in \{i\text{の類似度上位}N\text{アイテム}\}} (s_{i,j} * R_{u,j})}{\sum_{j \in \{i\text{の類似度上位}N\text{アイテム}\}} (|s_{i,j}|)} \quad \text{(式 4.2)}$$

また，類似アイテム数 N の値は，4.3 項に示す MovieLens 1M Dataset を用いて次のように決定した. データセットの timestamp の値の小さいもの 80%を学習用データ，残りの値の大きいもの 20%をテスト用データとしたとき，学習済みの本モデルによるレーティングの予測値を， MAE(Mean Absolute Error，平均絶対誤差)によって評価した．評価結果より，N の値として MAE が最も小さくなる 12 を採用した.

4.3 使用したデータセット

評価実験では，Grouplens による MovieLens 1M Dataset[3] をデータセットとして用いた．MovieLens Dataset は，映画のレビューサイトにおける，ユーザが映画につけたレーティングのデータセットである．データセットの一部を表 4.1 に示す．レーティングの値は最低 1 から最高 5 の間の 5 段階の値を持つ. なお，本データセットにおけるユーザ数は 6,040 人，映画の数は 3,706 本，レーティング数は 1,000,209 件である．

表 4.1 MovieLens 1M Dataset における レーティングデータの一部

userId	movieId	rating	timestamp
1	1193	5	978300760
1	661	3	978302109
1	914	3	978301968
1	3408	4	978300275

4.4 評価方法

評価は以下の手順によって行った．

1. 4.3 項で示したデータセットの全データを用いて，4.2 項に示した協調フィルタリングによる推薦モデルを学習させる．
2. 過去にユーザuが各アイテムに与えたレーティングを表すベクトルを$\boldsymbol{u}$としたとき，uがアイテムiに与えるレーティング$P_{u,i}$を出力する協調フィルタリングモデル$f_i(\boldsymbol{u}) = P_{u,i}$によって，全ユーザの全未評価アイテムに対する予測レーティングの算出を行う．
3. ユーザuにおける最も予測レーティングの高いアイテムiを推薦アイテムとして，レーティング予測$f_i(\boldsymbol{u}) = P_{u,i}$の説明を LIME によって生成する．
4. LIME によって出力された，特徴量jの重要度を$w_{LIME,j}(j = 1, \dots, n)$と表す．また，説明に用いた特徴量の集合を$I_{LIME} = \{j | w_{LIME,j} \neq 0\}$とする．協調フィルタリングモデル$f_i$における特徴量$j$の重要度$w_{CF,j}(j = 1, \dots, n)$は，アイテム$i$とアイテム$j$の類似度$s_{i,j}$を用いて式 4.3 のように表す．

$$w_{CF,j} = \begin{cases} s_{i,j} & (j \in \{i\text{の類似度上位}N\text{アイテム}\}) \\ 0 & (j \notin \{i\text{の類似度上位}N\text{アイテム}\}) \end{cases} \quad \text{(式 4.3)}$$

また出力値の計算に用いられたアイテムの集合を$I_{CF} = \{j | w_{CF,j} > 0\}$とする．ここで，LIME による説明$I_{LIME}$と$w_{LIME,j}$，協調フィルタリングの内部の解釈による説明$I_{CF}$と$w_{CF,j}$を複数の評価指標を用いて比較する．

5. LIME と協調フィルタリングのパラメータを変化させて 1~4 を繰り返す．変化させたパラメータの一覧を表 4.2 に示す．

表 4.2 実験で変化させたパラメータ

記号	説明	実験に用いた値
K	LIME の説明に用いる特徴量の数．I_{LIME}の要素数と一致する．アルゴリズム 1 参照.	1, 2, 4, 8, 12
S	LIME でサンプリングするベクトルの数．アルゴリズム 1 参照.	100, 200, …, 2000

4.5 評価指標

4.4 項に示した手順における評価指標には，Precision, Recall, nDCG を使用した．Precision, Recall は特徴量の重要度を考慮しない評価指標，nDCG は特徴量の重要度を考慮した評価指標である．

Precision, Recall の計算には，I_{LIME}，I_{CF}を用いて，それぞれ式 4.4，式 4.5 のように表す．Precision は，LIME の説明で用いられた特徴量のうち，実際の推薦モデルで出力の算出に用いられた特徴量の数の割合を表す．Recall は，実際の推薦モデルで出力の算出に用いられた特徴量のうち，LIME の説明で用いられた特徴量の数の割合を表す．

$$Precision = \frac{|I_{LIME} \cap I_{CF}|}{|I_{LIME}|} \quad \text{(式 4.4)}$$

$$Recall = \frac{|I_{LIME} \cap I_{CF}|}{|I_{CF}|} \quad \text{(式 4.5)}$$

nDCG(normalized Discounted Cumulative Gain)[12]は，ランキング予測で用いられる評価指標である．DCG_pは，上位p件のランキングの精度を表す指標であり，i番目に提示された項目の適合度rel_iを用いて式 4.5 で表される．$nDCG_p$は，ランキング予測が完全に適合した場合のDCG_pである$IDCG_p$によってDCG_pを正規化した値であり，式 4.6 で表される．本実験では，LIME による重要度の計算でi番目に重要度の高かった特徴量の，協調フィルタリングにおける実際の重要度$w_{CF,i}$を適合度rel_iとして$nDCG_p$を算出した．なお，pの値は，LIME の説明に用いる特徴量の数Kの値に応じて変更し，$nDCG_K$として評価を行った．

$$DCG_p = \sum_{i=1}^{p} \frac{rel_i}{\log_2(i+1)} = \sum_{i=1}^{p} \frac{w_{CF,i}}{\log_2(i+1)} \quad \text{(式 4.6)}$$

$$nDCG_p = \frac{DCG_p}{IDCG_p} \quad \text{(式 4.7)}$$

4.6 結果

本項では，表 4.2 に示したパラメータの変化による，説明の精度の評価結果，推薦の説明の生成にかかる計算時間，メモリ

[3] MovieLens | GroupLens https://grouplens.org/datasets/movielens/

使用量の測定結果を示す. 本実験を行った計算機の CPU は Intel Xeon CPU E5-2620 v4(2.10GHz), コア数 8, スレッド数 16, メモリは 128GB, OS は CentOS7.4 である.

4.6.1 精度

Precision, Recall, nDCG の測定結果をそれぞれ図 4.1, 図 4.2, 図 4.3 に示す.

いずれの評価指標でも, サンプリング数Sの値を上げると, LIME による説明の精度が上昇することが確認できる. 図 4.1 では, 説明に用いる特徴量の数Kの値を小さくすることでより Precision の値が高くなることが分かる. また, $K = 12$において, $S \geq 1800$で Precision$= 1$となり, 予測値に関与する特徴量が LIME によって完全に特定できていることが分かる. 図 4.2 では, Recall の値は, 実際の推薦モデルで出力の算出に用いられた特徴量の割合であることから, $K/12$が Recall の取りうる最大値となるため, Kの値が小さいと, Recall の値は小さくなる. Kの値を小さくすることは, ユーザに提示する説明がシンプルになる一方で, 完全な説明を与えることができなくなるという側面を持つことが分かる. 図 4.3 では, Precision と同様, Kの値を小さくすることでより nDCG の値が高くなることが分かる. また, $K = 12$において, $S \geq 1800$で $nDCG_{12} = 1$となり, 予測値に関与する特徴量の重要度も LIME によって完全に特定できていることが分かる.

4.6.2 計算時間

特定ユーザへの特定アイテムの推薦理由の説明 1 件の生成にかかる計算時間の測定結果を図 4.4 に示す. 図 4.4 より, 計算時間はサンプリング数Sが増えることで増加することが分かる. つまり, 説明の精度と計算時間はトレードオフの関係にあることが分かる. また, 説明に用いる特徴量の数Kの値は, 計算時間に影響しないことが分かる. 図 4.3 に示した, nDCG の測定の結果における $nDCG_{12}$=1 となった, $K = 12, S = 1800$での計算時間は, 2.10 秒であり, ユーザへの説明の提示を考慮する上では現実的な時間である.

また, LIME のアルゴリズムにおける計算時間のかかっている部分を特定するために, アルゴリズムの特定の部分における計算時間の測定を行った. 具体的には, アルゴリズム 1 の 6 行目にあたる, サンプリングしたベクトルとその出力結果から, K個の特徴量を用いた線形回帰モデルを選定する部分(K-Lasso)の計算時間の測定を行った. $K = 12$における, 特定ユーザへの特定アイテムの推薦理由の説明 1 件の説明の生成にかかる合計計算時間と, K-Lasso にかかる計算時間の比較を図 4.5 に示す. 説明生成にかかる時間から K-Lasso にかかる時間を引いた値は, K-Lasso 以外の部分にかかった時間, すなわちアルゴリズム 1 の 2〜5 行目にあたる, N個の入力ベクトルのサンプリングを行い, モデルの出力を計算する時間である. 図 4.5 より, K-Lasso とそれ以外の部分の計算時間がそれぞれ同程度の計算時間を持ち, どちらもサンプリング数Sの値に依存していることが分かる. 従って, さらに説明の生成の計算時間を短縮するためには, Sをより少なくするための, 効率的な部分へのサンプリング方法を検討する必要があると考えられる.

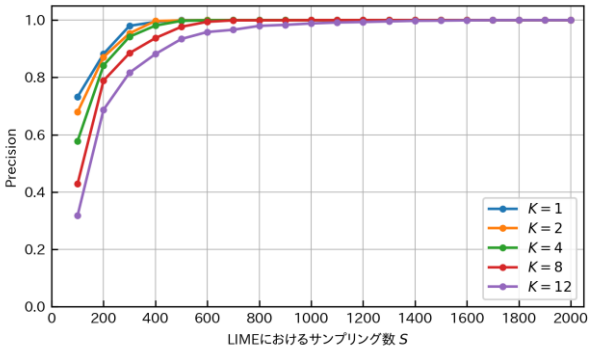

図 4.1 推薦結果の説明における Precision の測定結果

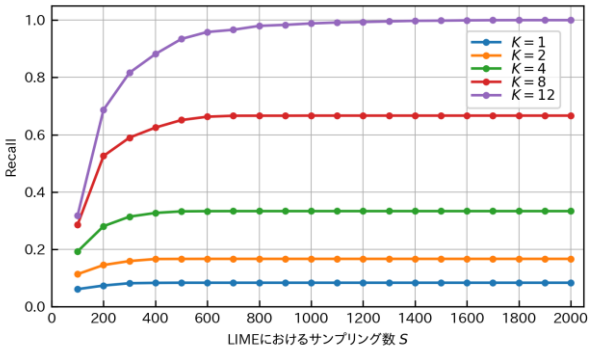

図 4.2 推薦結果の説明における Recall の測定結果

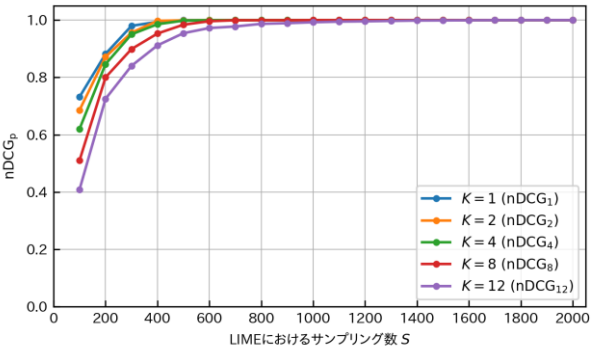

図 4.3 推薦結果の説明における nDCG の測定結果

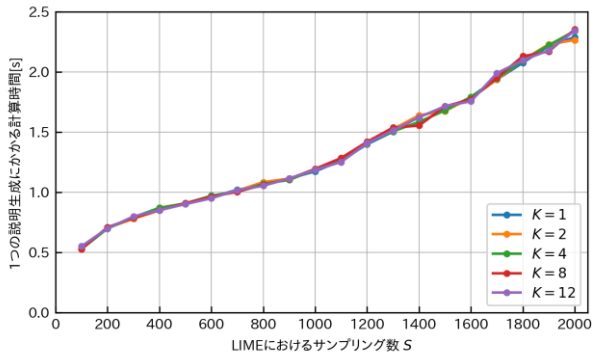

図 4.4 推薦結果の説明にかかる計算時間の測定結果

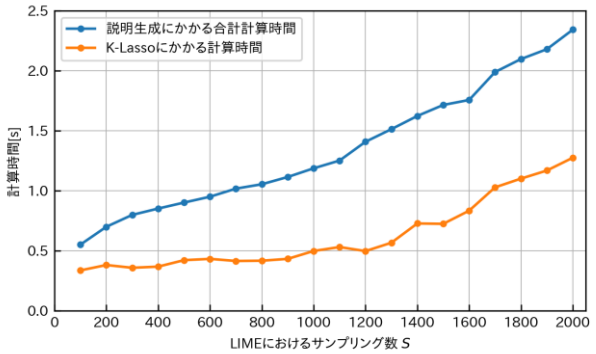

図 4.5 説明の生成にかかる合計計算時間と K-Lasso 部分の計算時間の測定結果 ($K = 12$)

4.6.3 メモリ使用量

本実験で用いた推薦理由の説明を生成する関数における，特定ユーザへの特定アイテムの推薦理由の説明 1 件の説明の生成時の最大メモリ使用量の測定結果を表 4.3 に示す. 表 4.3 より，メモリ使用量はサンプリング数Sが増えることで増加することが分かる. $S = 1000$における最大メモリ使用量は 120.1MB であるが，実際のサービスにおける複数の推薦理由の説明の生成を同時に行う環境を想定すると，使用量の値は大きく，削減への検討が必要であると考えられる.

表 4.3 LIME による説明を生成する関数における最大メモリ使用量の測定結果 ($K = 12$)

LIME における サンプリング数 S	最大メモリ使用量 [MB]
100	23.6
500	95.1
1000	120.1
1500	132.0
2000	174.2

4.7 本手法を用いた説明の例

協調フィルタリングに対して本手法を適用して得られた結果を用いて，ユーザに提示する推薦理由の説明の一例を作成した．今回は，MovieLens データセットにおける userId=1 のユーザに対して，協調フィルタリングによる予測レーティングが最も高い映画 movieId=3101 の推薦説明を LIME のパラメータを $S = 2000, K = 4$として特徴量の重要度を予測し，得られた特徴量を用いて説明を作成した．作成した説明の一例を図 4.6 に示す.

あなたへのおすすめの映画 “Fatal Attraction (1987)” は，
以下のあなたの過去の評価に基づいています.

“Rain Man (1988)”	★★★★★
“Schindler’s List (1993)”	★★★★★
“Apollo 13 (1995)”	★★★★★
“Back to the Future (1985)”	★★★★★

図 4.6 協調フィルタリングモデルに提案手法を適用した場合の説明の例

5. おわりに

本論文では，近年の機械学習を用いた推薦システムにおける説明手法が，推薦モデルのアルゴリズムに依存する点，及び解釈性を高めるために精度を下げるような手を加える必要がある点を解決する手法を提案した．具体的には，任意の機械学習モデルに対して，精度を下げずに解釈が可能な LIME と呼ばれる手法を用いて推薦モデルを解釈し，ユーザに対して推薦理由を提示する方法を提案した．評価実験では，協調フィルタリングに対する LIME による説明が，説明に用いる特徴量の数$K = 12$, サンプリング数 $S = 1800$における Recall，$nDCG_{12}$の値がどちらも 1，計算時間が 2.10 秒であり，現実的な計算時間で重要度の高い特徴量を示した説明が生成可能であることを示した.

本手法を使用した研究の今後の予定としては，1)特徴量の重要度から説明インタフェースを自動生成する手法の検討，2)近年提案されているニューラルネットワークなどを用いた推薦モデルに対する本手法の適用，及び評価方法の検討，3)実サービスの環境を想定した，説明の生成の省メモリ化の検討，4)実際の推薦システムへの本手法の適用によるユーザによる説明の評価の実施，などを検討している.

参考文献

[1] J. L. Herlocker, J. A. Konstan, and J. Riedl, “Explaining Collaborative Filtering Recommendations,” in Proc. of ACM CSCW, pp. 241–250, 2000.

[2] S. Seo, J. Huang, H. Yang, and Y. Liu, “Interpretable Convolutional Neural Networks with Dual Local and Global Attention for Review Rating Prediction,” in Proc. of ACM RecSys, pp. 297–305, 2017.
[3] B. Abdollahi and O. Nasraoui, “Using Explainability for Constrained Matrix Factorization,” in Proc. of ACM RecSys, pp. 79–83, 2017.
[4] M. T. Ribeiro, S. Singh, and C. Guestrin, “‘Why Should I Trust You?’ Explaining the Predictions of Any Classifier,” in Proc. of ACM KDD, pp. 1135–1144, 2016.
[5] Y. Zhang, G. Lai, M. Zhang, Y. Zhang, Y. Liu, and S. Ma, “Explicit factor models for explainable recommendation based on phrase-level sentiment analysis,” in Proc. of ACM SIGIR, pp. 83–92, 2014.
[6] B. Abdollahi and O. Nasraoui, “Explainable Restricted Boltzmann Machines for Collaborative Filtering,” in Proc. of ICML Workshop WHI, 2016.
[7] X. He, T. Chen, M.-Y. Kan, and X. Chen, “TriRank: Review-aware Explainable Recommendation by Modeling Aspects,” in Proc. of ACM CIKM, pp. 1661–1670, 2015.
[8] R. Heckel, M. Vlachos, T. Parnell, and C. Duenner, “Scalable and Interpretable Product Recommendations via Overlapping Co-Clustering,” in Proc. of IEEE ICDE, pp. 1033–1044, 2017.
[9] C. Chen, M. Zhang, Y. Liu, and S. Ma, “Neural Attentional Rating Regression with Review-level Explanations,” in Proc. of WWW, pp. 1583–1592, 2018.
[10] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl, “Item-based collaborative filtering recommendation algorithms,” in Proc of WWW, pp. 285–295, 2001.
[11] L. Zheng, V. Noroozi, and P. S. Yu, “Joint Deep Modeling of Users and Items Using Reviews for Recommendation,” in Proc of ACM WSDM, pp. 425–434, 2017.
[12] K. Järvelin and J. Kekäläinen, “Cumulated gain-based evaluation of IR techniques,” ACM Trans. Inf. Syst., vol. 20, no. 4, pp. 422–446, Oct. 2002.