DEIM2022 C24-6

推薦システムにおけるレビュー文の特性と PPLM を用いた説明文生成モデル

小久保彰博† 杉山 一成†

† 京都大学大学院情報学研究科 〒 606-8501 京都市左京区吉田本町
E-mail: †kokubo@db.soc.i.kyoto-u.ac.jp, ††kaz.sugiyama@i.kyoto-u.ac.jp

あらまし 推薦システムにおける説明可能性は，次第に推薦システムの研究において不可欠な側面となってきている．説明可能な推薦とは，ユーザに対してアイテムを推薦するだけではなく，なぜそのアイテムが推薦されたのかに関する説明も提示する推薦のことである．本研究では，レビューの特性を生かして学習の際のレビューの有用性をモデ ル化することが，ユーザからアイテムへの評価値の予測のタスクだけでなく，その説明を生成するタスクに対しても有効であることを示し，事前学習済みの言語モデルを用いて生成された説明が定量的にも定性的にも品質の高いものであることを示す．

キーワード 推薦システム，説明可能性，文生成

1 はじめに

近年，インターネット上には情報が氾濫している．そのため，ユーザの興味や関心に適合する情報が埋もれないように，ユーザに対して適切な情報を提供する推薦システムはますます必要とされている．

最近では，推薦システムはその推薦の精度だけでなく，推薦の説明可能性にも注目が集まっている．なぜなら，ユーザにアイテムが推薦された理由を説明をすることによって，推薦システムの透明性，説得力，有効性，信頼性，ユーザの満足度が向上する [1] からである．このような推薦は一般に，説明可能な推薦（Explainable Recommendation）と呼ばれる．

さて，推薦の精度を高めるために，レビューをうまく用いて推薦システムを構築する研究 [2] [3] が報告されている．これらの研究は，豊富に存在するレビューからユーザとアイテム間の情報を抽出できる点に着目している．しかし，すべてのレビューが有用であるとは限らない [2]．そこで，レビューの有用性を推定することで推薦性能を向上する研究 [4] [5] もある．さらに，レビューの特性を生かした研究 [6] もある．レビューの特性は，ユーザやアイテムに関して多くの洞察をもたらす．これらの観点は，Explainable Recommendation の研究ではまだ試されておらず，本研究ではレビューの特性を活かして提案モデルを構築する．

推薦モデルの説明の手法やフォーマットはいろいろ提案されており, [1] ではそれらの手法を 6 つに分類している．中でも推薦モデルが説明を文章の形式でユーザに提供する方法は，“Textual sentence explanation” と呼ばれる．説明を文章で示す場合，事前に定義されたテンプレートを用いて文章を構成する場合 [7] と自然言語モデルによって直接生成する場合 [8] [9] [10] とある．テンプレートを用いた手法の問題点として説明が単純すぎてしまう点や繰り返しばかりになってしまう点があり，ユーザが商品を購入するのに十分な説得力や表現力がない．これに対して，言語モデルを学習して説明を生成する方法はより自由で表現力の高い説明を生成できる．この場合，モデルの主な目的は，ユーザが過去に書いたレビューに似た文章を生成することである．これらの研究の前提には，レビュー文は，ユーザがなぜその商品を買ったのか，その商品を買った後にどんなことを感じたのかも説明できるため，潜在的に説明性を持っているという考え方がある．これらの手法の主な欠点は，テキストの説明を生成するために，ゼロから言語モデルを学習することである．モデルの学習には，かなりの計算コストを必要とする．また，それらの言語モデルの学習に用いられるレビュー文の集合は，そのほか最先端の言語モデルが学習に用いる膨大なコーパスと比較するとサイズが小さい．これが原因で，生成される説明の流暢性に課題が残る．

そこで，BERT [11] や GPT-2 [12] のような事前に学習された言語モデルを，好みのデータセットに合わせてファインチューニングする転移学習の手法を，説明可能な推薦に用いる研究が提案されている [13]．説明を生成するモジュールは，事前学習済みの巨大な言語モデルの上に構築されており，転移学習 [14] により学習時間と学習すべきパラメーターの個数ともに少なく，学習に要するコストの観点で非常に効率的である．また，それと同時に，高品質でドメインに特化した説明が提供できる．しかし, [13] には，その内部での extractive な操作において改善できる部分が残されている．本研究ではこの点に対し改善を行い，正解の文章により近い説明を生成することに成功した．

本研究の貢献は，つぎのようにまとめられる．

1. 評価値予測のタスクと説明生成のための文脈予測のタスクを同時に学習する multi-task learning を行なうことのできる推薦モデルを設計した．この推薦モデルはレビューの特性を用いて推薦の精度と説明生成の精度を向上する．この観点は explainable recommendation では，まだ試されていない．

2. 文脈を予測するタスクを導入することによって，説明を

生成するモジュールでの extractive な操作を改善した．PPLM への入力となる conditional text を，アテンションによって決定するのではなく，直接的に説明文の文脈を予測することによって決定することで，より正解の文に近い説明文を生成することができる．

3. モデルの性能を評価値予測の観点と説明生成の観点から定量的に評価した．評価値予測では MSE と MAE を算出し，モデルの性能を評価した．説明生成の評価には，自然言語処理の分野で広く用いられる BLEU [15] や ROUGE [16] のスコア，文章の多様性を評価する Distinct のスコア，生成した文章と正解の文章との Pearson Correlation Coefficient を用いた．

4. 生成された説明文を定性的に評価した．

2 関 連 研 究

2.1 評価値の予測

評価値を予測をする推薦モデルはこれまでに多く提案されている．ここではレビューを用いた研究に注目する．Deep Co-operative Neural Network (DeepCoNN) [3] は初期に提案された手法の一つで，レビューのテキストからアイテムの属性やユーザの行動を学習して，ユーザからアイテムへの評価値を予測する深層学習のモデルである．DeepCoNN はレビューの独立性を強く仮定しているのに対して，Neural Attentive Rating Regression (NARRE) [2] は，レビュー文書の中で個々のレビューの分布を学習するアテンション機構によって，この仮定を改善した．Review Properties-based Recommendation Model (RPRM) [6] は，レビューのテキスト情報とそれに関連するレビューの特性の両方を利用することに着目した研究である．具体的には，レビューの長さ，時間，感情などの特性と，ユーザーの好みとの関係を活用することで，推薦モデルがレビューの有用性をより正確に学習でき，より効果的な推薦ができることを示した．

2.2 説明の生成

2.2.1 テンプレートベースの手法

あらかじめ決められた説明文のテンプレートをいくつか定義して，そのテンプレートに異なる単語を入れることでパーソナライズする手法はテンプレートベースと言われる．Explicit Factor Model (EFM) [7] は，Latent Factor Models (LFM) を用いて推薦精度を維持しながら,「あなたは，この製品の良い/悪い [機能] に興味があるかもしれません」とユーザに伝えることで，説明可能な推薦を行う．DEAML (Deep Explicit Attentive Multi-View Learning) [17] は Microsoft Concept Graph を用いる．テンプレートをベースとした説明の主な問題点は，同じことを繰り返しがちになり，それゆえに説得力に欠ける点がある．さらに，ユーザが過去に書いたレビューのスタイルとかけ離れるという点もある．

2.2.2 自然言語処理ベースの手法

多くの生成モデルは，説明可能な推薦を行うために言語モデルを学習している．Neural Rating Regression (NRT) [8] は Gated Recurrent Units (GRU) [18] を用いて，tips を生成する．ここで tips とは，レビューよりも簡潔な文章でユーザの体験や感じたことを表現したもののことである．Neural Template (NETE) [9] は，GRU を改善し，推薦の説明文で言及してほしいトピックを受け取り，それに合わせて柔軟に文章を生成する GFRU を提案した．PErsonalized Transformer for Explainable Recommendation (PETER) [10] は，ユーザとアイテムの ID から説明文の文脈予測をするタスクと説明文を生成するタスクを同時に学習することにより，Transformer の出力を個人最適化した．これらのモデルの弱点は，モデルの学習に必要となる時間とコストである．推薦の説明を生成するという条件付きの言語モデルをはじめから学習するのは，非常に大きな学習コストが必要となる．また，ユーザが書いたレビュー文はコーパスとして膨大とはいえず，ドメインに固有のデータのみで学習するため，生成される文章の品質は十分ではなく流暢さに問題が残る．そこで，EXplainable Recommendation using Plug and Play Language Model (ReXPlug) [13] では，説明文の生成に Plug and Play Language Model (PPLM) [14] を用いてこの問題に対処している．

3 提 案 手 法

3.1 問 題 設 定

ユーザ u がアイテム i をレビューした時の評価値を $r_{(u,i)}$ とする．また，ユーザ u が書いたレビューの集合を $\mathcal{D}_u$，アイテム i について書かれたレビューの集合を $\mathcal{D}_i$ とする．訓練データにある u と i の組 (u,i) の集合を Ω で表す．訓練データに含まれるすべてのユーザ u とアイテム i の組 (u,i) について，評価値 $r_{(u,i)}$ が存在し，かつ，レビュー文 $s_{(u,i)}$ が存在する．レビュー文は，k 個の特性の集合によっても表現される．

$$\mathbf{P} = \{P_1, P_2, \ldots, P_k\} \tag{1}$$

ユーザ u の，レビューの特性 P_1 への好みを捉えるために，ユーザ u のレビューの集合に含まれるそれぞれのレビューに対して，P_1 のスコアを算出する．

$$P_{1,u} = \left\{p_{1,1}, p_{1,2}, \ldots, p_{1,|\mathcal{D}_u|}\right\} \tag{2}$$

$p_{1,t}(t = 1, \cdots, |\mathcal{D}_u|)$ はユーザ u の t 番目のレビューの特性 P_1 のスコアである．$|\mathcal{D}_u|$ はユーザ u に関するレビューの集合の要素数である．特性のスコアは $[0,1]$ に写像される．提案するモデルは，上記のようにユーザが投稿したレビューの特性を活かすことによって，推薦の精度を向上するだけでなく，より良い説明を生成することを目指す．ユーザやアイテムの情報をモデル化する際に，レビューの特性を考慮することが，ユーザの好みやアイテムの属性の表現を豊かにすると考えられる．

本研究で提案する推薦モデルが取り組むタスクは，1) ユーザ u とアイテム i の組に対して評価値の予測を行い，$\hat{r}_{(u,i)}$ を出力すること，2) その上で，アイテム i がなぜユーザ u に推薦されたのかを正当化するような説明 $\hat{s}_{(u,i)}$ を，自然言語の文章によって出力すること，の 2 点である．

3.2 モデルの全体像

本研究で提案する手法は，次の 4 つのモジュールから構成される（図 1）．(a) はレビューの特性を考慮するモジュールで，ユーザが生成したレビューとその特性を生かすことによって，評価値の予測の精度や生成する説明の精度の向上を目指す．(b) は評価値を予測するモジュールで，レビューの潜在ベクトルと，ユーザとアイテムの識別用の埋め込みベクトルとを組み合わせ，ユーザからアイテムへの評価値を予測する．(c) は文脈を予測するモジュールで，ユーザ u にアイテム i を推薦したときの説明の潜在表現 $s_{(u,i)}$ を学習する．(d) は説明を生成するモジュールで，識別器を学習させたのちに，PPLM を利用することによって，推薦の評価値に適した説明を生成する．

3.3 (a): レビューの特性を考慮するモジュール

このモジュールは，レビューとその特性に注目するためのモジュールである．どのレビューの特性が，各レビューの有用性を表現するのに役立つかを学習する．本研究で提案するモデルでは，ユーザとアイテムをモデル化する際に，同じ構造のネットワークを用いる（図 1 (a)）．モジュールは，3 つのステップから構成され，以下でそれぞれについて説明する．

(a-1): Review Property Encoding

まずはユーザに関してのレビューとアイテムに関してのレビューそれぞれについて，意味的な情報をまとめるために，Universal Sentence Encoder [19] を用いて，埋め込みベクトルに変換する．各レビューは固定長 T のベクトルに変換される．次に，内積によって，レビューの表現ベクトルとレビューの各特性をエンコードする．各レビューの特性は，標準化した特性スコアの配列として表される．このスコアによって，異なるレビューに注目することができ，関連するレビューの特性についての知識をエンコードすることができる．例えば，レビュー文の長さという特性をエンコードすることによって，異なる長さが推薦の結果にどのように影響するのかについて，モデルが捉えることができるようになる．

あるレビュー文の特性 P_1 をエンコードする処理は，次のように表すことができる．

$$O_{u,P_1} = \left[x_1 p_{1,1}, x_2 p_{1,2}, \ldots, x_{|\mathcal{D}_u|} p_{1,|\mathcal{D}_u|}\right] \tag{3}$$

ただし，$x_{1,\ldots,|\mathcal{D}_u|}$ は，ユーザ u のレビュー文の埋め込み表現で，$p_{1,t}$ はユーザ u の t 番目のレビューの特性スコアで，$|\mathcal{D}_u|$ は，ユーザ u によるレビュー文の集合の要素数である．k 個のレビューの特性について，ユーザ u のレビュー文の集合をエンコードすると，以下のようになる．

$$O_u = [O_{u,P_1}, \ldots, O_{u,P_k}] \tag{4}$$

本研究では，レビューの特性として，以下の 4 つの特性を用いる．1) Age: レビューがどのくらい新しいかを表す特性である．2) Length: レビューがどのくらい長いかを表す特性である．3) Rating: レビューに関連するアイテムへの評価値を表す特性である．4) Helpful: レビューが他のユーザにとって有益であるかどうかを表す特性である．

(a-2): CNN Text Processing

レビュー文の埋め込み表現を特性スコアとともにエンコードした後，CNN を適用して，各レビューの表現を得る．これは，レビュー文をもとにした深層学習を用いる他の研究 [3] [2] [6] においても，よく使われている手法である．畳み込み層は，m 個のニューロンから構成されているとする．j 番目のニューロンはレビューの埋め込みベクトルに対して以下のように畳み込みを適用して特徴 z_j を出力する．

$$z_j = \mathrm{ReLU}\left(V * K_j + b_j\right) \tag{5}$$

ただし，V は入力となる長さ T のレビューのベクトル，K_j は j 番目のニューロンのフィルター，$*$ は畳み込み演算，b_j は重みである．活性化関数として，ReLU 関数を用いて特徴 z_j を得る．各ニューロン j は特徴 z に対して，max pooling 関数を用いてサイズ t の sliding window を適用する．すなわち，$z_1, z_2, \ldots, z_j^{(T-t+1)}$ を，sliding window 上で j 番目のニューロンが生成した特徴とすると，このニューロンの最終的な出力 o_j は，以下のようになる．

$$o_j = \max\left(z_1, z_2, \ldots, z_j^{(T-t+1)}\right) \tag{6}$$

各レビューに対して，m 個のニューロンからの出力を連結することによって，この層の最終的な出力を得る．これは以下のように表すことができる．

$$O = [o_1, o_2, \ldots o_m] \tag{7}$$

(a-3): Review Property Attention

(a-1) では，ユーザとアイテムの埋め込み表現を，レビューの異なる特性を考慮して埋め込み表現を変換した．(a-3) の主な目的は，どのレビューの特性が，ユーザの好みやアイテムの属性を表現するのに，より役に立つのかを見極めることである．ユーザとアイテムはそれぞれ，サイズが k のレビューの特性の重みベクトル ϕ_u と ϕ_i を持つ．k は用いるレビューの特性の個数である．あるユーザ u について，この層での変換は，以下のように定義される．

$$O'_u = \frac{\sum_{t=0}^{k} \phi_{u,t} O_{u,P_t}}{k} \tag{8}$$

3.4 (b): 評価値を予測するモジュール

このモジュールでは，モジュール (a) で算出されたレビューの潜在ベクトルと，ユーザとアイテムの識別用の埋め込みベクトルとを組み合わせ，ユーザ u のアイテム i への評価値を予測する．推薦はこの予測された評価値に基づいて行われる．ユーザ u のアイテム i への最終的な予測値 $\hat{r}_{(u,i)}$ は，以下のように算出される．

$$\hat{r}_{(u,i)} = \left(O'_u \oplus V_u\right) \odot \left(O'_i \oplus V_i\right) \tag{9}$$

ただし，$\oplus$ は結合演算で，レビューの埋め込みベクトル O' と識別用の埋め込みベクトル V を結合させる．$\odot$ はユーザ u とアイテム i の潜在ベクトル同士の要素ごとの積を表し，これによってユーザ u のアイテム i への評価値の予測 $\hat{r}_{(u,i)}$ を算出

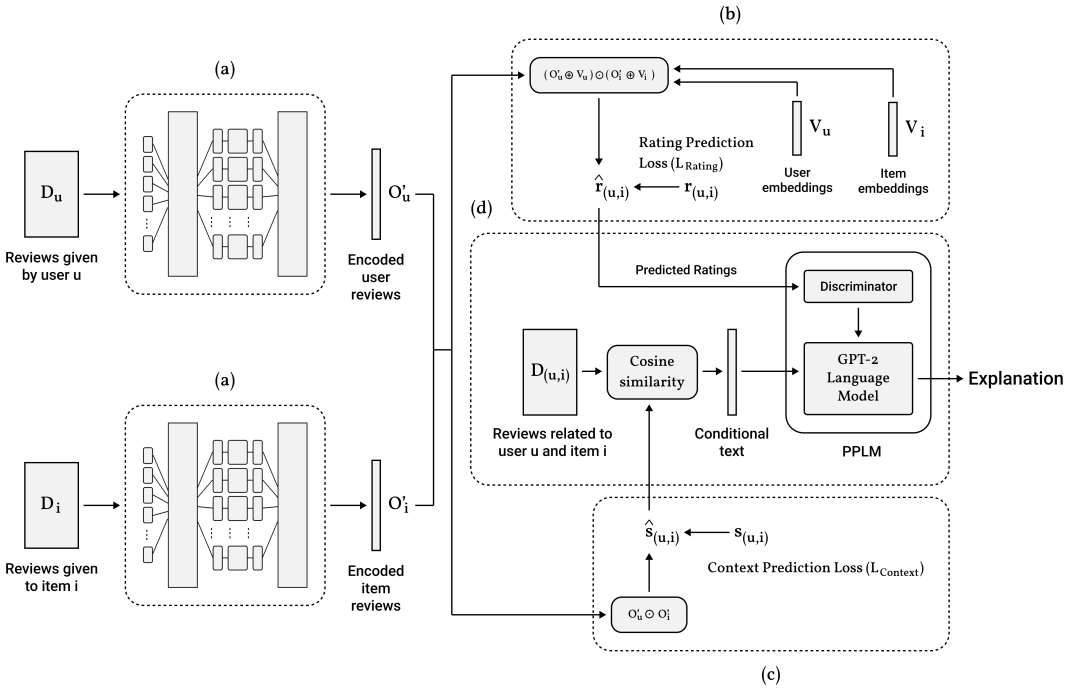

図 1　モデルの全体像

する．このタスクでは，損失関数として，Mean Square Error (MSE) を用いる．よって，評価値の予測の損失関数は以下のようになる．

$$\mathcal{L}_{Rating}(\Theta) = \frac{1}{|\Omega|} \sum_{(u,i)\in\Omega} \left(r_{(u,i)} - \hat{r}_{(u,i)}\right)^2 \qquad (10)$$

ただし，$r_{(u,i)}$ は評価値の正解で，Ω はデータセットである．

3.5　(c): 文脈を予測するモジュール

ここでのタスクは，ユーザ u とアイテム i の説明の潜在表現 $s_{(u,i)}$ を学習することである．評価段階で，このモジュールの出力は PPLM への入力である conditional text を決定する際に説明の文脈として用いられる．ここでのタスクのもう一つの目的は，正解の説明の表現 $s_{(u,i)}$ を目指すことによって，より豊かなユーザとアイテムの潜在表現を特定し，より精度の高い評価値の予測につなげることである．すなわち，評価値を予測するタスクと文脈を予測するタスクを同時に multi-task learning によって学習することで，それぞれのタスクの結果を向上させることも目的の一つである．

このモジュールの出力は，O'_u と O'_i の要素ごとの積で，

$$\hat{s}_{(u,i)} = O'_u \odot O'_i \qquad (11)$$

である．損失関数は MSE を用いる．よって，このモジュールの損失関数は以下のように定義される．

$$\mathcal{L}_{Context}(\Theta) = \frac{1}{|\Omega|} \sum_{(u,i)\in\Omega} \|s_{(u,i)} - \hat{s}_{(u,i)}\|^2 \qquad (12)$$

ただし，$s_{(u,i)}$ は，ユーザ u とアイテム i に対する説明の正解となる潜在表現で，Ω はデータセットである．

3.6　(d): 説明を生成するモジュール

このモジュールは，推薦モデルの評価段階で用いられるモジュールである．このモジュールは Plug and Play Language Model (PPLM) [14] をもとにしており，学習済みの識別器を用いて説明文を生成する．PPLM は attribute と conditional text をもとに，属性に沿ったテキストを出力する言語モデルである．このモジュールでは，次の 2 つの入力を受け取る．1) 予測した評価値 $\hat{r}_{(u,i)}$: PPLM への attribute として用いる．これはモジュール (b) の出力である．2) 予測した説明文の文脈 $\hat{s}_{(u,i)}$: conditional text を決定するために用いる．これはモジュール (c) の出力でる．

評価段階では，訓練データセット Ω に含まれないユーザ u とアイテム i の組 (u,i) に対して，モジュール (b) で評価値 $\hat{r}_{(u,i)}$ を予測し，モジュール (c) で説明の文脈 $\hat{s}_{(u,i)}$ を予測する．得られた評価値 $\hat{r}_{(u,i)}$ と説明の文脈 $\hat{s}_{(u,i)}$ を用いて，モジュール (d) で説明としてのテキストを出力する（図 1）．ここでは，この時に PPLM への入力となる conditional text の選び方を説明する．モジュール (c) で説明の文脈をすでに予測しており，この予測に最も近いレビュー文を推薦の対象となるユーザとアイテムに関連するレビュー文の中から選ぶ．これらのレビュー文はモデルの訓練時に Universal Sentence Encoder によってエンコードされたものである．また，予測した文脈との近さはコサイン類似度で算出するものとする．よって，ユーザ u とアイテム i に関連するレビュー文の集合を $D_{(u,i)}$ とすると，**conditional text**$_{(u,i)}$ は以下のように選ばれる．

$$\textbf{conditional text}_{(u,i)} = \underset{d\in D_{(u,i)}}{\operatorname{argmax}} \cos\left(\hat{s}_{(u,i)}, d\right) \qquad (13)$$

3.7 モデルの学習

モデル学習時の全体の損失関数は，次式のように各モジュールでの損失関数の線型結合として定義する．

$$\mathcal{L} = \lambda_{Rating} * \mathcal{L}_{Rating} + \lambda_{Context} * \mathcal{L}_{Context} \quad (14)$$

ただし，λ_{Rating} と $\lambda_{Context}$ は，ハイパーパラメータで，正の実数である．

4 実験の方針

実験は実データを用いて行ない，提案モデルを最新の研究で提案されている 2 つのモデルと比較する．また，multi-task learning の効果と説明文の文脈予測の効果について，着目する．そのため以下の Research Questions (RQs) に取り組む．

RQ1: 提案手法は，3 つの実データセットで行う評価値予測のタスクでベースラインを上回るのか

RQ2: 提案手法で用いるレビュー文の特性のうち，どの特性が評価値予測の性能に寄与しているのか

RQ3: 提案手法は，3 つの実データセットで行う説明文生成のタスクでベースラインを上回るのか

RQ4: 提案手法によって生成される文章は，推薦対象のユーザに最適化され，ユーザの好みとアイテムの属性を捉えることができているのか

RQ5: 提案手法でのマルチタスク学習は，評価値予測のタスクと説明文生成のタスクにおいてそれぞれのシングルタスク学習の性能を上回るのか

4.1 データセット

本研究では，Amazon の 5-core データセット [1]を用いて実験を行う．このデータセットは実際のオンラインシステム（amazon.com）から得られた実データである．Amazon のデータセットのうち，Digital Music，Video Games，Clothing の 3 つのカテゴリを用いる．これらはデータセットに含まれるインタラクションの総数やデータの density が異なる．表 1 にそれぞれのデータセットの統計情報を示す．

データセットは，80%が学習データ，10%がバリデーションデータ，10%がテストデータとなるように分割する．この時，先行研究 [6] にならい，各ユーザのインタラクションに対して同様のデータ分割の割合を適用してデータセット全体のデータ分割を行う．

4.2 評価尺度

用いる評価尺度には，評価値予測の精度を評価するためのものと，説明生成の精度を評価するものがある．

4.2.1 評価値予測の評価

評価値の予測の評価尺度として，Mean Square Error (MSE) と Mean Absolute Error (MAE) を用いる．これら二つの評価尺度は推薦システムの評価値予測のタスクに広く用いられているものであり，値が小さいほど良い評価値であることを示す．

1：http://jmcauley.ucsd.edu/data/amazon/index_2014.html

表 1　データセットの統計情報

Dataset	Reviews	Users	Items	Density
Digital Music	64.7K	5.5K	3.6K	0.3273%
Video Games	231.8K	24.3K	10.7K	0.0894%
Clothing	278.7K	39.4K	23.0K	0.0307%

そのことを，第 5 章での実験結果の表では，MSE↓ や MAE↓ のように表記する．

4.2.2 説明生成の評価

説明生成の評価として，BLEU [15] と ROUGE [16] を用いる．これらはそれぞれ，自然言語処理の分野の翻訳のタスクと要約のタスクで，生成された文章と正解の文章との近さを評価する際に広く用いられる評価尺度である．大きいほど良い評価値であるため，実験結果の表では，B-1 ↑のように表記する．

Distinct [20] は文章の多様性を評価するためのもので，Distinct-n は，文中のユニークな n-gram の数を計算することによって多様性の度合いを表す．説明文全体のコーパスに関する多様性を Global Distinct，説明文それぞれに注目した多様性を Local Distinct として以下のように定義する．

$$\textbf{Global Distinct}_n(C) = \frac{\textbf{Unique}_n(C)}{\textbf{Count}_n(C)} \quad (15)$$

$$\textbf{Local Distinct}_n(C) = \frac{1}{|C|} \sum_{s \in C} \frac{\textbf{Unique}_n(s)}{\textbf{Count}_n(s)} \quad (16)$$

ただし，C はコーパスで，$\textbf{Unique}_n(C)$ はコーパス C の中のユニークな n-gram の総数，$\textbf{Count}_n(C)$ はコーパス C の中の n-gram の総数を表す．値が大きいほど良い評価値であり，そのことを実験結果の表では，GD-1 ↑のように表記する．

PCC は，生成された説明文と正解の文の相関係数を計算したもので，生成した説明文と正解の文との意味上の類似性を評価するための尺度である [13]．生成された説明文と正解の文はそれぞれ，RoBERTa [21] によってエンコードされた上で PCC が算出される．RoBERTa のモデルは，STS ベンチマーク [22] という文章間の意味上の類似性に関するタスクに対して学習されたモデルである．値が大きいほど良い評価値であることを示し，そのことを，実験結果の表では PCC ↑のように表記する．

4.3 ベースラインの手法

RPRM [6]: レビューの特性を用いてユーザとアイテムのインタラクションを学習する推薦モデルである．これはランキングのための推薦モデルとして提案されており，推薦の説明文を出力しない．

ReXPlug [13]: cross attention network を用いて評価値の予測と説明文の生成を行う説明可能な推薦モデルで，説明文の生成の際には PPLM を用いる．

SR (Ours): Single_task Rating の略である．Rating Prediction のタスクに関して Single_task Learning をしたモデルの評価値予測の結果を SR と表記することにする．

MR (Ours): Multi_task Rating の略である．Multi_task Learning で学習したモデルで評価値予測した結果を MR と表記することにする．

SC (Ours): Single_task Contexts の略である．Context Prediction に関して Single_task Learning で学習したモデルによる Conditional Texts の結果を SC と表記することにする．

MC (Ours): Multi_task Contexts の略である．Multi_task Learning で学習したモデルによる Conditional Texts の結果を MC と表記することにする．

ME (Ours): Multi_task Explanations の略である．Multi_task Learning で学習したモデルによる説明文の出力結果を，ME と表記することにする．

5 実験結果

5.1 RQ1: 提案手法の評価値予測における評価

表 2 に，評価値予測の結果を示す．まず，RPRM と MR (Ours) を比較する．Digital Music，Video Games，Clothing の 3 つのデータセット全てにおいて，MSE と MAE ともに，MR (Ours) が RPRM を上回っている．RPRM は説明可能な推薦モデルではなく，単なる推薦モデルである．MR (Ours) は評価値の予測のタスクと同時に，説明文の文脈予測のタスクも同時に行いモデルを学習している．文脈予測の学習が評価値の予測のタスクの結果を良くすると考えることができる．

次に，ReXPlug と MR (Ours) を比較する．Digital Music と Video Games では MR (Ours) の方が ReXPlug を上回っている．一方で，Clothing では ReXPlug の方が良い結果が出ている．Digital Music と Video Games に比べて，Clothing はデータ数が大きく，Density が小さいデータセットである．評価値予測のタスクにおいては，MR (Ours) は比較的小さく密なデータセットに対しては有効である一方で，データセットが大きくスパースなデータセットに対しては有効ではないと考えることができる．

5.2 RQ2: レビュー文の特性に関する Ablation Study

表 3 に，ablation study の結果を示す．表中の MR (Ours) は，全ての特性，すなわち，age, length, rating, helpful を用いてマルチタスク学習を行ったモデルでの評価値予測の結果を表す．また，- age は，4 つのレビュー文の特性のうち，特性 age のみを除いた 3 つの特性を用いてマルチタスク学習を行ったモデルでの評価値予測の結果を表す． ここでは，結果が最も悪いモデル，すなわち，MSE，MAE の値が最も大きくなったモデルは，その特性が MR (Ours) での結果に一番影響を及ぼす特性であると考える．

表 3 の結果から，評価値予測のタスクに対して効果的なレビューの特性は，rating と helpful であったと言える．それに対して，length と age は結果に対する効果が比較的小さい．今回の提案モデルについては，いつそのレビューがなされたのか，あるいはそのレビューの長さは長いのか短いのかは，それほど重要ではなかったと考えることができる．

5.3 RQ3: 提案手法の説明生成における定量的な評価

表 4 に，説明生成における定量的な評価結果を示す．まず，BLUE の結果に注目する．ME (Ours) と ReXPlug を比較すれば一貫して ME (Ours) の結果が良い．BLEU の観点においては提案モデルの方がより正解に近く，質の高い文を出力できる，ということがわかる．また，Digital Music のデータセットではほとんど違いが現れなかったが，SC (Ours) と MC (Ours) を比較すると，MC (Ours) の方がおおむね良い結果となった．これは，文脈予測のタスクをシングルタスクとして学習するよりも，評価値予測のタスクと合わせてマルチタスク学習した方がより良い結果が得られることを示す．さらに，Clothing のデータセットには当てはまらないが，MC (Ours) と ME (Ours) を比較すると，ME (Ours) の方がおおむね良い結果となった．これは，PPLM により説明文を生成することが，BLEU の観点でより正解に近づいたと考えることができる．

表 2　評価値予測の結果

Dataset	Digital Music		Video Games		Clothing	
Model	MSE ↓	MAE ↓	MSE ↓	MAE ↓	MSE ↓	MAE ↓
RPRM	1.0447	0.7621	1.3883	0.8866	1.4222	0.9036
ReXPlug	1.0197	0.7976	1.3598	0.9534	**1.1763**	**0.8881**
SR (Ours)	1.0255	0.7566	1.3531	0.8832	1.3968	0.8922
MR (Ours)	**0.9880**	**0.7461**	**1.3416**	**0.8748**	1.3836	0.8961

表 3　レビューの特性に関する Ablation Study の結果

Dataset	Digital Music		Video Games		Clothing	
Model	MSE ↓	MAE ↓	MSE ↓	MAE ↓	MSE ↓	MAE ↓
MR (Ours)	0.9880	0.7461	1.3416	0.8748	1.3836	0.8961
- age	1.0152	0.7516	1.3418	0.8632	1.3854	0.8922
- length	1.0177	0.7508	1.3437	0.8681	1.3922	0.8931
- rating	**1.0214**	**0.7613**	1.3507	0.8719	**1.3965**	**0.9011**
- helpful	1.0186	0.7559	**1.3615**	**0.8772**	1.3947	0.8990

次に ROUGE の結果に注目する．まず，ME (Ours) と ReXPlug を比較すれば，一貫して ME (Ours) の方が ROUGE の値が大きく，良い結果が得られている．よって，提案モデルの方が ROUGE の観点において，より正解に近く，質の高い文章を出力できると結論づけることができる．また，SC (Ours) と MC (Ours) を比較すると，一貫して SC (Ours) の結果の方が良くなった．ROUGE の観点では，文脈予測のタスクをシングルタスクで学習する方が，評価値予測のタスクと合わせてマルチタスク学習するアプローチよりも良い結果となると言える．さらに，MC (Ours) と ME (Ours) を比較すると，Digital Music の ROUGE-2-f を除いたすべてにおいて，ME (Ours) の方が良い ROUGE の値を出力している．これは，PPLM により説明文を生成することが，ROUGE の観点でより正解に近づいていることを示すと考えることができる．

Distinct の結果に注目すると，Global Distinct でも Local Distinct でも概ね ReXPlug が良い結果を出すことがわかった．これは，出力される文章の多様性は ReXPlug が一番大きいことを示す．一方で，提案手法も GD も ReXPlug の値と比較して著しく小さいとは言えないため，明らかな問題につながる結果ではないと考えられる．

最後に，PCC の結果に注目する．ME (Ours) と ReXPlug とを比較すると，3 つのデータセットすべてにおいて ME (Ours) が大きく ReXPlug を上回っている．これは提案モデルの方が

表 4　生成された説明文の定量的な評価結果

Dataset	Model	B-1 ↑	B-2 ↑	B-3 ↑	B-4 ↑	R-1-f ↑	R-2-f ↑	R-l-f ↑	GD-1 ↑	GD-2 ↑	LD-1 ↑	LD-2 ↑	PCC ↑
Digital Music	ReXPlug	17.2708	6.0042	2.0242	0.7353	15.0722	1.3469	13.7083	**0.0310**	0.1996	**0.6745**	**0.9254**	0.3639
	SC (Ours)	25.2341	10.8742	4.8879	2.4998	19.0371	**2.6398**	17.1215	0.0296	**0.2335**	0.6311	0.9237	0.5615
	MC (Ours)	25.5558	10.8645	4.8383	2.4669	18.9320	2.4621	17.0719	0.0234	0.1735	0.6173	0.9214	**0.5638**
	ME (Ours)	**26.9629**	**11.2011**	**4.9349**	**2.5007**	**19.2409**	2.4421	**17.3864**	0.0270	0.1840	0.6191	0.9196	0.5567
Video Games	ReXPlug	15.5806	5.9042	1.9634	0.6514	16.0934	1.5191	14.5698	**0.0142**	0.1158	**0.6888**	0.9372	0.3297
	SC (Ours)	22.2484	9.6805	3.8321	**1.6267**	19.4274	**2.6694**	17.4299	0.0126	**0.1215**	0.6351	0.9360	**0.5824**
	MC (Ours)	24.4232	10.2827	3.9064	1.5298	19.3611	2.4722	17.3987	0.0121	0.1109	0.6503	**0.9418**	0.5453
	ME (Ours)	**25.2410**	**10.4369**	**3.9076**	1.5271	**19.6834**	2.4809	**17.7109**	0.0122	0.1068	0.6429	0.9383	0.5417
Clothing	ReXPlug	17.1042	5.6037	1.7340	0.5993	15.1643	0.9713	13.8196	**0.0130**	**0.1242**	0.7678	0.9481	0.3301
	SC (Ours)	20.8574	8.0979	3.1817	**1.4839**	**19.7861**	**2.0931**	**17.7862**	0.0121	0.1190	0.7709	0.9563	**0.5640**
	MC (Ours)	**21.4451**	**8.2176**	**3.2083**	1.4789	19.4223	1.9607	17.4509	0.0115	0.1128	**0.7767**	**0.9579**	0.5394
	ME (Ours)	20.1462	7.8353	3.0564	1.3882	19.6618	2.0256	17.7347	0.0102	0.1025	0.7365	0.9502	0.5382

表 5　生成された説明文 (explanation) と正解の文 (ground truth) の比較

(a)	explanation	These **sweatpants** are great except for the lack of **pockets**.
	ground truth	I love these **sweatpants**, they fit nice and are very comfortable. I just wish they had some **pockets** in them.
(b)	explanation	I really like these **gloves**. They're long, they fit my wide hands well, and they **stay warm when sweat**. **Great price** and a great product.
	ground truth	These **gloves** are exactly what I wanted. They come at a **great price** and are well worth the money I spent on them. And they **stay warm when I sweat**! Great buy.
(c)	explanation	If you have **large thighs**, these **socks** will **roll down**. But they are **cute** and sexy. Also very warm to wear. I love these **socks** and will be wearing these in the future.
	ground truth	These **socks** are **cute** with boots; however, if your have **thick thighs** these **socks** will **roll down** to your knee. I wear them folder at the knee to avoid rolling down.
(d)	explanation	This game is **neverending**. My son loves it, you can do so much in it. It goes on **for ever**. Now I see why it took them so long to release this game.. The **graphics** are excellent.. I highly recommend **Grand Theft Auto V** for the Xbox 360.
	ground truth	I purchased this game for my son and he loves it. the **graphics** are amazing and its **endless** to play. So much open enivironment to go to. I would recommend **Grand Theft Auto V** if your a fan of the series.

ReXPlug よりも文章の意味的により正解に近い説明を生成することができることを示す．SC (Ours) と MC (Ours) とを比較すると，Digital Music においては PCC のスコアについて，僅差で MC (Ours) が SC (Ours) を上回ったが，Video Games と Clothing では SC(Ours) の方が大きかった．PCC の値に対しては，文脈予測のタスクをシングルタスクとして学習する方が，より良いスコアとなることがわかる．MC (Ours) と ME (Ours) とを比較すると，3つのデータセット全てについて，やや MC (Ours) が ME (Ours) を上回っている．これによって，PPLM で文章の sentiment を補正することにより，PCC のスコアがやや小さくなることがわかる．

5.4　RQ4: 提案手法の説明生成における定性的な評価

表 5 に，モデルが生成した説明文と対応する正解の文をまとめた．同じアイテムや同じ意味を表す表現は，太文字にされている．例えばケース (a) に注目すると，推薦対象のアイテムであるスウェットパンツについて言及できており，かつ説明文と正解の文章の感情が揃っている．そしてスウェットパンツにポケットがない点についても言及できており，事実関係が適切に把握され述べられている．説明は個人最適化されており，未知の正解が言及する同じアイテムについて，同じトピックから説明がなされている．提案した推薦モデルでは，レビューの特性とその内容を抽出してユーザの好みやアイテムの属性を捉え，文脈予測のタスクによって高品質な説明を生成できることがわかる．

5.5　RQ5: 提案モデルにおけるマルチタスク学習の効果

評価値予測のタスクについては，表 2 からわかる通り，Multi_task Learning の効果があると言える．これは，MR と SR の結果を比較するとわかる．評価値予測のタスクだけに最適化するのではなく，同時に説明の文脈予測のタスクでモデルを学習することで，ユーザの好みやアイテムの属性についてよりよく捉えることができ，推薦の汎用性が上がると考えることができる．

説明文生成のタスクについては，概ね効果がある傾向にあるが，評価指標やデータセットによって効果があるとは言えない場合もあった．BLUE では，マルチタスク学習が結果をよくすることがわかった．ROUGE では，マルチタスク学習が結果をよくする傾向にある一方で，Clothing のような dentisy が小さいデータセットではシングルタスク学習の方が良い結果を出した．Distinct では，GD でも LD でも大きな差があるとは言えなかった．PCC では，文脈予測のタスクをシングルタスク学習した方が結果がよくなった．

以上から，評価値予測のタスクについては Multi_task Learn-

ing の効果があり，文脈予測のタスクについては，効果は限定的であると結論づけられる．

6 おわりに

本研究では，これまで説明可能な推薦には用いられていなかったレビュー文の特徴を利用して推薦モデルの構築に取り組んだ．また，説明文の文脈を予測するというタスクを設定し，その結果を直接的に用いて conditional texts を決定することにより，PPLM の出力がより高品質となるようなアーキテクチャを設計した．さらに，評価値予測のタスクと文脈予測のタスクをマルチタスク学習として同時に学習することを提案した．また，既存研究では実際のユースケースに合わない方法で実験を行っていたのに対し，本研究では時間軸に沿ってデータセットを学習データ，バリデーションデータ，テストデータに分割した．テストデータに含まれるユーザ数とデータセットに含まれるユーザ数とを一致させることによって，推薦が行えないユーザがないように実験を行った．実際の Amazon の 3 つのデータセットによる実験で，評価値の予測のタスクは概ね既存研究に匹敵する，またはそれを上回る結果を出し，説明文の生成のタスクでは既存研究を上回る結果を出した．説明の正解の設定の仕方や，説明の定量的な評価の方法は確立されておらず，さらなる研究が必要である．

文　献

[1] Yongfeng Zhang and Xu Chen. Explainable recommendation: A survey and new perspectives. *arXiv preprint arXiv:1804.11192*, 2018.

[2] Chong Chen, Min Zhang, Yiqun Liu, and Shaoping Ma. Neural attentional rating regression with review-level explanations. In *Proceedings of the 2018 World Wide Web Conference (WWW '18)*, pp. 1583–1592, 2018.

[3] Lei Zheng, Vahid Noroozi, and Philip S Yu. Joint deep modeling of users and items using reviews for recommendation. In *Proceedings of the 10th ACM International Conference on Web Search and Data Mining (WSDM '17)*, pp. 425–434, 2017.

[4] Konstantin Bauman, Bing Liu, and Alexander Tuzhilin. Aspect based recommendations: Recommending items with the most valuable aspects based on user reviews. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '17)*, pp. 717–725, 2017.

[5] Xinyu Guan, Zhiyong Cheng, Xiangnan He, Yongfeng Zhang, Zhibo Zhu, Qinke Peng, and Tat-Seng Chua. Attentive aspect modeling for review-aware recommendation. *ACM Transactions on Information Systems (TOIS)*, Vol. 37, No. 3, pp. 1–27, 2019.

[6] Xi Wang, Iadh Ounis, and Craig Macdonald. Leveraging review properties for effective recommendation. In *Proceedings of the Web Conference (WWW '21)*, pp. 2209–2219, 2021.

[7] Yongfeng Zhang, Guokun Lai, Min Zhang, Yi Zhang, Yiqun Liu, and Shaoping Ma. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR '14)*, pp. 83–92, 2014.

[8] Piji Li, Zihao Wang, Zhaochun Ren, Lidong Bing, and Wai Lam. Neural rating regression with abstractive tips generation for recommendation. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17)*, pp. 345–354, 2017.

[9] Lei Li, Yongfeng Zhang, and Li Chen. Generate neural template explanations for recommendation. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management (CIKM '20)*, pp. 755–764, 2020.

[10] Lei Li, Yongfeng Zhang, and Li Chen. Personalized transformer for explainable recommendation. *arXiv preprint arXiv:2105.11601*, 2021.

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT '19)*, pp. 4171–4186.

[12] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, Vol. 1, No. 8, p. 9, 2019.

[13] Deepesh V Hada and Shirish K Shevade. Rexplug: Explainable recommendation using plug-and-play language model. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21)*, pp. 81–91, 2021.

[14] Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. Plug and play language models: A simple approach to controlled text generation. *arXiv preprint arXiv:1912.02164*, 2019.

[15] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL '02)*, pp. 311–318, 2002.

[16] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Proceedings of the ACL-04 Workshop on Text Summarization Branches Out*, pp. 74–81, 2004.

[17] Jingyue Gao, Xiting Wang, Yasha Wang, and Xing Xie. Explainable recommendation through attentive multi-view learning. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI-19)*, pp. 3622–3629, 2019.

[18] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[19] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. Universal sentence encoder for english. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP '18): System Demonstrations*, pp. 169–174, 2018.

[20] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A diversity-promoting objective function for neural conversation models. *arXiv preprint arXiv:1510.03055*, 2015.

[21] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[22] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.