

# 推薦システムにおける説明文生成モデルの提案

小久保彰博<sup>†</sup> 杉山 一成<sup>†</sup> 吉川 正俊<sup>†</sup>

<sup>†</sup> 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: <sup>†</sup>kokubo@db.soc.i.kyoto-u.ac.jp, <sup>††</sup>{kaz.sugiyama,yoshikawa}@i.kyoto-u.ac.jp

**あらまし** 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。説明可能な推薦の評価は、推薦の精度と説明の良さで評価される。推薦の精度は、上位  $n$  件にどの程度適合するアイテムが含まれるかという尺度で定量的に評価される一方で、説明の良さは定性的に評価する研究が多い。これまでの説明可能な推薦システムの多くは、グラフベースの手法が提案されているが、推薦の説明の部分をグラフのパスとして出力して、そのパスが定性的に妥当であることを具体例と共に提示するにとどまっているという問題点がある。そこで本研究では、説明の良さを定量的に評価することを目的として、アイテムを推薦する根拠となる文を生成するモデルを提案する。

**キーワード** 推薦システム, 説明可能性, 文生成

## 1 はじめに

インターネット上の情報が増え続けている現在、大量のアイテムの中からユーザが興味をもつようなアイテムをおすすめする推薦システムはますます重要な役割をになっている。推薦システムはEコマースや様々なジャンルのアプリケーションで活用されており、具体的には動画のストリーミングサービスである YouTube<sup>1</sup> や Netflix<sup>2</sup>, 音楽のストリーミングサービスである Apple Music<sup>3</sup> や Spotify<sup>4</sup>, Eコマース上でアイテムを推薦する Amazon<sup>5</sup> など使われている。推薦手法においては、協調フィルタリング [1][2] や行列因子分解 [3][4], 深層学習 [5][6][7] など、これまでに様々なモデルが提案され、推薦精度が改善され続けている。また、アイテムを推薦する際に説明をつけることによって、各ユーザの興味や関心に適合するアイテムをより選択しやすくなり、推薦したアイテムに対して説得力が増し、その推薦システムに対する信頼性や、ユーザの満足度に繋がるということが報告されている [8][9]。

## 2 関連研究

### 2.1 説明の仕方

推薦の説明の仕方には、様々なものがある。協調フィルタリングを用いる推薦では、ターゲットとなるユーザと似たようなユーザが高く評価していることを示す手法 [10] や、ユーザが高く評価したアイテムと推薦するアイテムが似ていることを示す手法 [11] がある。コンテンツベースの推薦では、ユーザと関連している特徴を学習しその特徴をもつアイテムを推薦した上で、その特徴を示す手法 [12] がある。また、Eコマースでのレ

ビューや SNS での投稿などユーザが生成した情報を用いて推薦する場合、ユーザのレビューにトピックモデル [13] を適用したものをアイテムの推薦理由として用いる手法 [14] や説明文を生成する手法 [15][16] がある。

### 2.2 グラフベースの手法

Wang ら [17] は、埋め込みを用いた手法の強みである一般化と木構造を用いたモデルの強みである説明可能性を掛け合わせた手法を提案した。

### 2.3 文生成の手法

テンプレートを用いた文生成の手法があり、これはあらかじめ説明文のテンプレートを用意しておきユーザごとに推薦理由となる特徴の部分だけを入れ替える手法である。Zhang ら [16] は特徴をテンプレートに当てはめ、Wang ら [18] は特徴と意見をテンプレートに当てはめて推薦理由の文章を生成した。

自然言語処理の技術をもとに、テンプレートを用いず直接文を生成する手法もある。Costa ら [19] は、long-short term memory (LSTM) [20] を用いて説明文を生成した。

Li ら [21] は、Eコマースサイトなどのウェブサービスでの星評価とともに *Tips* を生成する研究を行った。Yelp などのウェブサービスでは、ユーザがアイテム (Yelp の場合はお店) を評価する方法に新しい *Tips* という手段が設けられている。これまではユーザがアイテムを評価する際は 1~5 の数字で星評価するかレビュー文を書くかの選択肢があった。そこに新しく、*Tips* を書くという手段ができた。*Tips* とは、レビューよりも少ない言葉で簡潔にユーザの体験や感じたことを表すものである。*Tips* によってユーザはアイテムに関してすぐに判断をつけることができる。

Li ら [22] は、説明文生成フレームワークである Neural Template によって、安定して推薦の理由を生成する研究を行った。これによってテンプレート文を用いた手法よりも表現の幅が広がり、文章のスタイルがないまま文を生成する手法と比較して、

1 : <https://www.youtube.com/>

2 : <https://www.netflix.com/>

3 : <https://www.apple.com/apple-music/>

4 : <https://www.spotify.com/>

5 : <https://www.amazon.co.jp/>

説明の質を安定して向上させることが可能になった。

## 3 モデル

### 3.1 モデルの全体像

本研究の目標は、協調フィルタリングと同様にユーザーのアイテムに対するレビューの評価値と特徴を予測し、それに基づいてアイテムを推薦することと、予測した特徴を用いて推薦の説明として文章を生成することである。評価値と特徴の予測はモデルの左側のコンポーネントで行い、説明文の生成はモデルの右側のコンポーネントで行う。

### 3.2 レビュー評価値の予測

左側の部分のコンポーネントの目標は、ユーザーの潜在因子  $u$  とアイテムの潜在因子  $v$  を得ることである。レビュー評価値を予測するために、ここでは多層パーセプトロンをモデルとして用いる。

$U \in \mathbb{R}^{k_u \times m}$  をユーザーの潜在因子、 $V \in \mathbb{R}^{k_v \times n}$  をアイテムの潜在因子とする。ただし、 $k_u$  をユーザーの潜在因子の次元、 $k_v$  をアイテムの潜在因子の次元、 $m$  はユーザー数、 $n$  はアイテム数とする。また、 $\mathbf{W}_{uh}^r \in \mathbb{R}^{d \times k_u}$  と  $\mathbf{W}_{vh}^r \in \mathbb{R}^{d \times k_v}$  をそれぞれ重みとし、 $\mathbf{b}_h^r \in \mathbb{R}^d$  をバイアスとする。  $d$  は  $\mathbf{h}^r$  の次元とする。このとき、最初の潜在因子から隠れ層への写像は以下のようにできる。

$$\mathbf{h}^r = \sigma(\mathbf{W}_{uh}^r \mathbf{u} + \mathbf{W}_{vh}^r \mathbf{v} + \mathbf{b}_h^r) \quad (1)$$

ただし、活性化関数  $\sigma(\cdot)$  は、(2) 式で定義されるシグモイド関数とする。

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (2)$$

さらに層を追加することでより良い精度が期待され、以下のように隠れ層を追加する。  $l$  番目の隠れ層の重みを  $\mathbf{W}_{hh_l}^r \in \mathbb{R}^{d \times d}$  とすると、以下のようにできる。

$$\mathbf{h}_l^r = \sigma(\mathbf{W}_{hh_l}^r \mathbf{h}_{l-1}^r + \mathbf{b}_{h_l}^r) \quad (3)$$

隠れ層の数を  $L$  とすると、 $\mathbf{h}_L^r$  は最後の層の出力であり、この出力を評価値  $\hat{r}$  に変換する。  $\mathbf{W}_{hr}^r \in \mathbb{R}^d$  と  $b^r \in \mathbb{R}$  を用いて、以下のようにできる。

$$\hat{r} = \mathbf{W}_{hr}^r \mathbf{h}_L^r + b^r \quad (4)$$

レビュー評価値の予測を回帰問題として定義し、最適なユーザーの潜在因子  $U$  とアイテムの潜在因子  $V$  とパラメータ  $\Theta$  を求めるべく、以下のように損失関数を定義する。

$$\mathcal{L}^r = \frac{1}{2|\mathcal{X}|} \sum_{u \in \mathcal{U}, i \in \mathcal{I}} (\hat{r}_{u,i} - r_{u,i})^2 \quad (5)$$

ただし、 $\mathcal{X}$  は学習データセット、 $r_{u,i}$  はユーザー  $u$  がアイテム  $i$  を評価したときの正解データとする。

推薦の際は、同じユーザーに対して評価値の予測が高いアイテムから順に推薦する。

### 3.3 特徴の予測

図の左側のコンポーネントのもう一つの目標として、右側の説明文を生成するコンポーネントで用いる特徴をユーザーの潜在因子  $u$  とアイテムの潜在因子  $v$  から予測することがある。説明文で言及して欲しい特徴を予測するモデルとして、ここでは多層パーセプトロンをモデルとして用いる。

ユーザー  $u$  とアイテム  $i$  に対応する特徴を  $f_{u,i}$  とする。  $U \in \mathbb{R}^{k_u \times m}$  をユーザーの潜在因子、 $V \in \mathbb{R}^{k_v \times n}$  をアイテムの潜在因子とする。ただし、 $k_u$  をユーザーの潜在因子の次元、 $k_v$  をアイテムの潜在因子の次元、 $m$  はユーザー数、 $n$  はアイテム数とする。また、 $\mathbf{W}_{uh}^f \in \mathbb{R}^{d \times k_u}$  と  $\mathbf{W}_{vh}^f \in \mathbb{R}^{d \times k_v}$  をそれぞれ重みとし、 $\mathbf{b}_h^f \in \mathbb{R}^d$  をバイアスとする。  $d$  は  $\mathbf{h}^f$  の次元とする。このとき、最初の潜在因子から隠れ層への写像は以下のようにできる。

$$\mathbf{h}^f = \sigma(\mathbf{W}_{uh}^f \mathbf{u} + \mathbf{W}_{vh}^f \mathbf{v} + \mathbf{b}_h^f) \quad (6)$$

ただし、活性化関数  $\sigma(\cdot)$  は、(2) 式で定義されるシグモイド関数とする。

さらに非線形変換のための層を追加し、 $\mathbf{h}_L^f$  を隠れ層の出力とする。  $\mathbf{h}_L^f$  を次元  $|\mathcal{V}|$  のベクトル  $\hat{f}$  にマップする層も追加でき、以下のようにできる。

$$\hat{f} = \text{softmax}(\mathbf{W}_{hf}^f \mathbf{h}_L^f + \mathbf{b}^f) \quad (7)$$

ただし、 $\text{softmax}(\cdot)$  はソフトマックス関数で、 $\mathbf{W}_{hf}^f \in \mathbb{R}^{|\mathcal{V}| \times d}$  は重み、 $\mathbf{b}^f \in \mathbb{R}^{|\mathcal{V}|}$  はバイアスである。

$\hat{f}$  の中で値が一番大きい単語が特徴として選ばれる。すなわち、予測される特徴  $f^*$  は以下のように定義される。

$$f^* = \arg \max_{w_i \in \mathcal{V}} \hat{f}^{(w_i)} \quad (8)$$

モデルを学習するために、損失関数を *cross entropy loss* を用いて以下のように定義する。

$$\mathcal{L}^f = \frac{1}{|\mathcal{X}|} \sum_{u \in \mathcal{U}, i \in \mathcal{I}} -\log \hat{f}_{u,i}^{(w_{u,i})} \quad (9)$$

### 3.4 説明文の生成

近年、Long Short-Term Memory (LSTM) [20] や Gated Recurrent Unit (GRU) [23] のようなゲート付きニューラルネットワークは、テキストの生成タスクで良い結果を出している。また、テキストを生成する際に、最初の隠れ状態の他に特徴を入力として与える GRU を発展させた Gated Fusion Recurrent Unit (GFRU) [22] が提案されている。これによって生成する文章をこれまで以上にコントロールすることができる。今回はこの GFRU を用いて説明文を生成する。

説明文を生成する際、図の左側のコンポーネントで出力されたアイテムの評価値の予測を *sentiment* として用いる。具体的には、[1,5] である評価値  $\hat{r}_{u,i}$  に対して、3 未満だった場合は -1, 3 以上だった場合は +1 へと写像する。これによって  $s_{u,i} \in \mathbb{R}^d$  を得る。

多層パーセプトロンを用いて、ユーザーの潜在因子  $u$  とアイテム

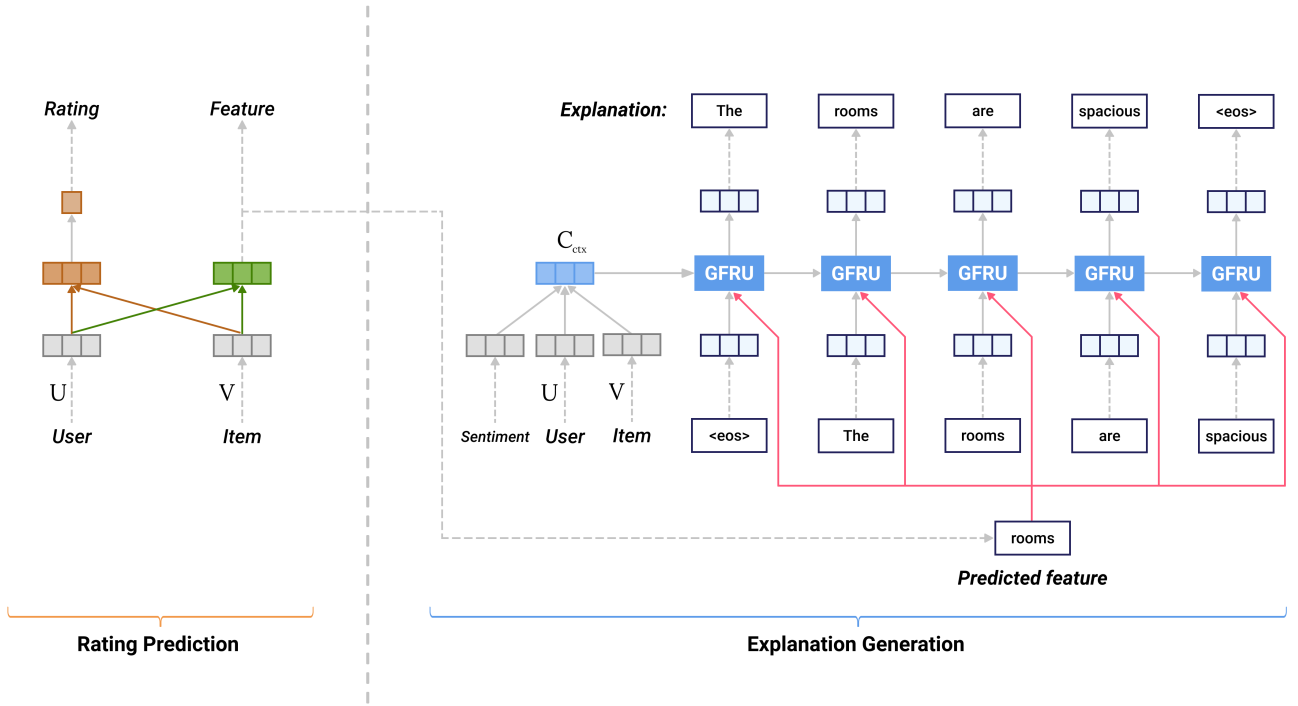


図1 モデルの全体像

ムの潜在因子  $v$  と *sentiment* を最初の隠れ層へと (10) 式を用いてエンコードする。

$$\mathbf{h}_0^s = \tanh(\mathbf{W}^s[\mathbf{u}, \mathbf{v}, s_{u,i}] + \mathbf{b}^s) \quad (10)$$

ただし,  $[\cdot]$  はベクトルの結合で,  $\mathbf{W}^s \in \mathbb{R}^{|\mathcal{V}| \times 3d}$  は重み,  $\mathbf{b}^s \in \mathbb{R}^{|\mathcal{V}|}$  はバイアス,  $\tanh(\cdot)$  は *hyperbolic tangent* 関数である。

エンコードされた  $\mathbf{h}_0^s$  はデコーダーの最初の隠れ状態として使われる。各ステップでの隠れ状態は再帰的に計算される。GFRU では,  $t$  ステップ目の隠れ状態は,  $t-1$  ステップ目の隠れ状態と,  $t-1$  ステップ目の出力である単語の分散表現と, 各ステップに入力される特徴の分散表現によって算出される。

$$\mathbf{h}_t^s = g(\mathbf{x}_{t-1}, \mathbf{h}_{t-1}^s, \mathbf{x}_t) \quad (11)$$

$g(\cdot)$  はデコーダーであり, ここでは GFRU である。

デコードの際, 以下の式にしたがって再帰的に単語が選ばれる。

$$p(x_t | x_{<t}, \mathbf{h}_0^s) = \text{softmax}_{x_t}(\mathbf{W}^s \mathbf{h}_t^s + \mathbf{b}^s) \quad (12)$$

ただし,  $\text{softmax}(\cdot)$  はソフトマックス関数で,  $\mathbf{W}^s \in \mathbb{R}^{d \times |\mathcal{V}|}$  は重み,  $\mathbf{b}^s \in \mathbb{R}^{|\mathcal{V}|}$  はバイアスである。  $x_{<t}$  はステップ  $t$  以前に生成された単語群で,  $x_t$  はステップ  $t$  で生成される単語である。各ステップごとに  $p(x_t | x_{<t}, \mathbf{h}_0^s)$  が一番大きい単語  $x_t$  が選ばれる。

説明文生成の学習の際, 損失関数は先行研究に従い *cross entropy loss* を用いる。すなわち以下のように定義される。

$$\mathcal{L}^s = \frac{1}{|\mathcal{X}|} \sum_{u \in \mathcal{U}, i \in \mathcal{I}} \frac{1}{|S_{u,i}|} \sum_{t=1}^{|S_{u,i}|} -\log p(x_t) \quad (13)$$

ただし,  $S_{u,i}$  はユーザ  $u$  とアイテム  $i$  に関する説明の正解データで,  $|S_{u,i}|$  はその文章の長さ,  $p(x_t)$  はモデルが予測した単語  $x_t$  の確率である。

### 3.5 GFRU

GFRU (Gated Fusion Recurrent Unit) [22] は, これまでのデコーダーの生成する説明が不安定になってしまうという欠点を補うために提案されたものである。生成する文章中で言及して欲しい単語を特徴として入力し, 文章をデコードする際に用いるというものである。

GFRU は3つのコンポーネントからなり, 2つの GRU と1つの GFU [24] からなる。説明文を扱う GRU と特徴を扱う GRU とそれらの出力を統合する GFU である。説明文を扱う GRU は一つ前のステップで出力された単語を入力とし, 特徴を扱う GRU は与えられた特徴を毎回のステップでの入力とする。

ステップ  $t$  を考える。GFRU の一つ前のステップの隠れ状態を  $\mathbf{h}_{t-1} \in \mathbb{R}^d$  として, 一つ前のステップの単語の出力を  $\mathbf{x}_{t-1} \in \mathbb{R}^d$  とする。ステップ  $t$  において, 説明文を扱う GRU の出力である  $\mathbf{h}_t^{GRU-s} = g^{GRU-s}(\mathbf{x}_{t-1}, \mathbf{h}_{t-1})$  は, 以下のように計算される。

$$\begin{cases} \mathbf{z}_t^{GRU-s} = \sigma(\mathbf{W}_z^{GRU-s}[\mathbf{x}_{t-1}, \mathbf{h}_{t-1}] + \mathbf{b}_z^{GRU-s}) \\ \mathbf{r}_t^{GRU-s} = \sigma(\mathbf{W}_r^{GRU-s}[\mathbf{x}_{t-1}, \mathbf{h}_{t-1}] + \mathbf{b}_r^{GRU-s}) \\ \mathbf{h}_t^{GRU-s} = \tanh(\mathbf{W}_h^{GRU-s}[\mathbf{x}_{t-1}, \mathbf{r}_t^{GRU-s} \odot \mathbf{h}_{t-1}] + \mathbf{b}_h^{GRU-s}) \\ \tilde{\mathbf{h}}_t^{GRU-s} = \mathbf{z}_t^{GRU-s} \odot \mathbf{h}_{t-1} + (1 - \mathbf{z}_t^{GRU-s}) \odot \tilde{\mathbf{h}}_t^{GRU-s} \end{cases} \quad (14)$$

ただし,  $\mathbf{W}_x^{GRU-s} \in \mathbb{R}^{n \times (d+n)}$  と  $\mathbf{b}_x^{GRU-s} \in \mathbb{R}^n$  はモデルのパラメータであり,  $\mathbf{r}_t^{GRU-s}$  と  $\mathbf{z}_t^{GRU-s}$  はそれぞれ過去の隠れ状態をどれほど無視するかとどれほど保持するかを決定するものである. 同じようにして, 一つ前のステップの GFRU の隠れ状態  $\mathbf{h}_{t-1} \in \mathbb{R}^n$  と, 与えられた特徴の単語  $\mathbf{x}_f \in \mathbb{R}^d$  を用いて, 特徴を扱う GRU の出力  $\hat{\mathbf{h}}_t^{GRU-f} = g^{GRU-f}(\mathbf{x}_f, \mathbf{h}_{t-1})$  は計算される.

次に, GFU を用いて  $\hat{\mathbf{h}}_t^{GRU-s}$  と  $\hat{\mathbf{h}}_t^{GRU-f}$  をステップ  $t$  の GFRU の出力である  $\mathbf{h}_t$  に統合する.

$$\begin{cases} \hat{\mathbf{h}}_t^{GRU-s} = \tanh(\mathbf{W}_{GRU-s} \mathbf{h}_t^{GRU-s}) \\ \hat{\mathbf{h}}_t^{GRU-f} = \tanh(\mathbf{W}_{GRU-f} \mathbf{h}_t^{GRU-f}) \\ k = \sigma(\mathbf{w}_k \hat{\mathbf{h}}_t^{GRU-s}, \hat{\mathbf{h}}_t^{GRU-f}) \\ \mathbf{h}_t = (1-k) \odot \hat{\mathbf{h}}_t^{GRU-s} + k \odot \hat{\mathbf{h}}_t^{GRU-f} \end{cases} \quad (15)$$

ただし,  $\mathbf{W}_{GRU-s} \in \mathbb{R}^{n \times n}$ ,  $\mathbf{W}_{GRU-f} \in \mathbb{R}^{n \times n}$  と  $\mathbf{w}_k \in \mathbb{R}^{2n}$  はモデルのパラメータである.

## 4 実験の方針

### 4.1 データセット

実験のデータセットには異なるドメインのサービスのデータを用いる. 具体的には, 旅行に関する口コミや価格を比較できるウェブサイトである TripAdvisor のデータセットと, レストランや飲食店の口コミ情報サイトである Yelp のデータセット, さらに, EC サイトである Amazon のデータセットを用いる.

### 4.2 評価指標

評価値の予測の評価指標として, Root Mean Square Error (RMSE) と Mean Absolute Error (MAE) を用いる. これら二つの評価指標は推薦システムの評価予測のタスクに広く用いられているものである. ユーザ  $u$  とアイテム  $i$  に対する評価値の予測を  $\hat{r}_{u,i}$  とし, 正解の評価値を  $r_{u,i}$  とし, データ数を  $N$  とすると,  $RMS E$  は以下のように計算される.

$$RMS E = \sqrt{\frac{1}{N} \sum_{u,i} (r_{u,i} - \hat{r}_{u,i})^2} \quad (16)$$

また, 同様にして  $MAE$  は以下のように計算される.

$$MAE = \frac{1}{N} \sum_{u,i} |r_{u,i} - \hat{r}_{u,i}| \quad (17)$$

アイテム推薦の精度の指標として, Normalized Discounted Cumulative Gain (NDCG) [25] と Hit Ratio (HR) を用いる.

生成した説明文を評価する指標として, 自然言語処理の要約のタスク評価で用いられる ROUGE [26] と翻訳のタスク評価で用いられる BLUE [27] を使用する.

## 5 おわりに

本研究では, 説明可能な推薦の説明文の良さを定量的に評価することを目的として, アイテムを推薦する根拠となる文の正解を用意し, 文生成を行い, 主に要約の分野で使われる指標を用いることによって定量的に評価する手法を提案した. 具体的

な実験は, 今後行う予定である.

## 文 献

- [1] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, and John Riedl. Grouplens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pp. 175–186, 1994.
- [2] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pp. 285–295, 2001.
- [3] Yehuda Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 426–434, 2008.
- [4] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, Vol. 42, No. 8, pp. 30–37, 2009.
- [5] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*, pp. 173–182, 2017.
- [6] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys (CSUR)*, Vol. 52, No. 1, pp. 1–38, 2019.
- [7] Lei Zheng, Vahid Noroozi, and Philip S Yu. Joint deep modeling of users and items using reviews for recommendation. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pp. 425–434, 2017.
- [8] Nava Tintarev and Judith Masthoff. Explaining recommendations: Design and evaluation. In *Recommender systems handbook*, pp. 353–382. Springer, 2015.
- [9] Yongfeng Zhang and Xu Chen. Explainable recommendation: A survey and new perspectives. *arXiv preprint arXiv:1804.11192*, 2018.
- [10] Jonathan L Herlocker, Joseph A Konstan, and John Riedl. Explaining collaborative filtering recommendations. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work*, pp. 241–250, 2000.
- [11] Behnoush Abdollahi and Olfa Nasraoui. Using explainability for constrained matrix factorization. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pp. 79–83, 2017.
- [12] Jesse Vig, Shilad Sen, and John Riedl. Tagsplanations: explaining recommendations using tags. In *Proceedings of the 14th international conference on Intelligent user interfaces*, pp. 47–56, 2009.
- [13] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research (JMLR)*, Vol. 3, pp. 993–1022, 2003.
- [14] Yao Wu and Martin Ester. Flame: A probabilistic model combining aspect based opinion mining and collaborative filtering. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, pp. 199–208, 2015.
- [15] Rose Catherine and William Cohen. Transnets: Learning to transform for recommendation. In *Proceedings of the eleventh ACM conference on recommender systems*, pp. 288–296, 2017.
- [16] 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*, pp. 83–92, 2014.
- [17] Xiang Wang, Xiangnan He, Fuli Feng, Liqiang Nie, and Tat-Seng Chua. Tem: Tree-enhanced embedding model for explainable recommendation. In *Proceedings of the 2018 World Wide Web Conference*, pp. 1543–1552, 2018.
- [18] Nan Wang, Hongning Wang, Yiling Jia, and Yue Yin. Explainable recommendation via multi-task learning in opinionated text data. In *The 41st International ACM SIGIR Conference on Research & De-*

velopment in *Information Retrieval*, pp. 165–174, 2018.

- [19] Felipe Costa, Sixun Ouyang, Peter Dolog, and Aonghus Lawlor. Automatic generation of natural language explanations. In *Proceedings of the 23rd International Conference on Intelligent User Interfaces Companion*, pp. 1–2, 2018.
- [20] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, Vol. 9, No. 8, pp. 1735–1780, 1997.
- [21] 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*, pp. 345–354, 2017.
- [22] 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*, pp. 755–764, 2020.
- [23] 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.
- [24] John Arevalo, Thamar Solorio, Manuel Montes-y Gómez, and Fabio A González. Gated multimodal units for information fusion. *arXiv preprint arXiv:1702.01992*, 2017.
- [25] Kalervo Järvelin and Jaana Kekäläinen. Ir evaluation methods for retrieving highly relevant documents. In *In Proc. of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 41—48. ACM New York, NY, USA, 2000.
- [26] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pp. 74–81, 2004.
- [27] 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*, pp. 311–318, 2002.