

複数の商品購買順序情報を考慮する 拡張 Translation-based Recommendation モデルの提案

情報数理応用研究

5220C030-9 李ア舒
指導教員 後藤正幸

A Proposal for an Extended Translation-based Recommendation Model Considering Sequential Purchased Product Information

YASHU Li

1. はじめに

近年、EC サイトを通じた購買行動が一般的なものになった一方、膨大な数の商品から適切な商品を探し出す時間が増加し、ユーザが嗜好に合致したアイテムを発見できない可能性も増大している。これに伴い、各顧客に対して適切な商品を提示する推薦システムは重要な役割を果たすようになった。また、EC サイトにおける各顧客の購買履歴は連続した購入商品の系列として与えられ、その購買傾向は顧客ごとに、過去の購買行動と顧客の嗜好、およびその直近の変化に影響されると考えられる。

一方、推薦システムの分野では、大量の購買行動履歴から顧客の特徴を表現する特徴空間を構築し、対象となるデータ（アイテム、ユーザなど）を任意の低次元ベクトルで表現する埋め込み表現モデルが幅広く活用されている。その中の1つの手法である Translation-based Recommendation Model (TransRec)[1] は、ユーザによる購入アイテムの連続的な傾向を捉え、各ユーザが最後に購入したアイテムとユーザの嗜好を用いて次の購入アイテムを予測する推薦モデルである。

しかし TransRec モデルでは、購買行動のモデル化において直前に購入されたアイテムのみを用いるため、(1) 以前購入された複数のアイテムの影響を考慮できない、(2) ユーザの突発的な嗜好変化を表現することができない、という2つの課題がある。(1) については、ユーザの購買行動は複雑であり、直前の購買以外にも影響を受ける可能性が高いと考えられる。そのため、より多くの商品情報を考慮することで、より適した推薦が可能と期待される。(2) については、ユーザが突発的な事由でアイテムを購入した際、ただ直前の1つアイテムからの推薦ではこの単発的な購買情報を重視してしまい、ユーザの嗜好を十分に反映することは難しいと考えられる。

以上より本研究では、直前に購入したアイテム1点だけでなく、 N 回前までの購入アイテムも含めて考慮することが可能な N -TransRec を提案する。この場合、ユーザの連続に購入したアイテムの前後関係を学習することが可能になる。最後に、提案手法の有効性を評価するために、実データを用いた実験を行い、提案手法の有効性を検証する。

2. 関連研究

2.1. 埋め込み表現を用いた推薦システム

埋め込み表現モデルとは、自然言語処理分野で使われる重要なベクトル表現技術である。その代表的なモデルとして、Word2Vec[2] が提案されている。この手法はニューラ

ルネットワークを用いて、大規模なテキストコーパスから単語間の関連性を学習するものである。具体的には全ての単語を同一特徴空間に埋め込み、それぞれの単語を異なるベクトルで表現している。さらに推薦システムに導入するため、Word2Vec をベースとして拡張した Item2Vec [3] が提案された。Item2Vec は単語の代わりにアイテムを、ユーザとアイテムの共起関係に基づいて潜在空間に埋め込むモデルである。しかし、これらの埋め込み手法ではユーザの購買行動における購入の前後関係の情報を考慮できず、連続して購買するユーザの嗜好を把握することが難しい。

2.2. Translation-based Recommendation

本研究では埋め込み表現であり、かつ購入されたアイテムの間の連続的な傾向を考慮できる TransRec に着目する。アイテムの集合を $I = \{I_1, I_2, \dots, I_M\}$ 、ユーザの集合を $U = \{u_1, u_2, \dots, u_V\}$ と定義する。TransRec では、全アイテムと全ユーザは K 次元の埋め込み空間 Φ にベクトルとして埋め込まれる。ここで、ユーザ $u \in U$ の嗜好ベクトル \vec{T}_u を、全ユーザの共通の嗜好ベクトル \vec{t} とユーザ固有の嗜好の方向性ベクトル \vec{t}_u の和で定義する。すなわち、ユーザの嗜好ベクトルは式 (1) で与えられる。

$$\vec{T}_u = \vec{t} + \vec{t}_u. \quad (1)$$

次に、ユーザ u が $j-1$ 番目に購入したアイテム r_{j-1}^u からその次に買うアイテム r_j^u への遷移関係を式 (2) で定義する。ここで、購入するアイテム r_j^u のベクトル \vec{r}_j^u は、特徴空間 Φ において $\vec{T}_u + \vec{r}_{j-1}^u$ とベクトル間の距離が近くなるように学習される。

$$\vec{T}_u + \vec{r}_{j-1}^u \approx \vec{r}_j^u. \quad (2)$$

次に購入するアイテム r_j^u の購買確率は、そのアイテムの人気度 β_j と、予測するアイテムベクトル $\vec{T}_u + \vec{r}_{j-1}^u$ 、および各アイテムベクトル \vec{r}_j^u との距離によって決定されるものとし、式 (3) のように購入確率を定義する。

$$\text{Prob}(r_j^u | u, r_{j-1}^u) \propto \beta_j - d(\vec{r}_j^u, \vec{r}_{j-1}^u + \vec{T}_u), \quad (3)$$

s.t. $\vec{r}_{j-1}^u, \vec{r}_j^u \in \Psi \subseteq \Phi, \forall r_{j-1}^u, r_j^u \in I.$

なお、 $d(\vec{x}, \vec{y})$ は \vec{x} と \vec{y} の距離を表し、本研究では L_2 距離を用いる。また、「次元の呪い」の問題を避けるために、 Ψ を特徴空間の原点からの距離が1以下の K 次元空間（単位球）を表すものとして、アイテムベクトルのサイズを1以下に制限している。

次にパラメータの更新について述べる。ユーザ $u \in \mathcal{U}$ の購買履歴 $S^u = S_1^u, S_2^u, \dots, S_{m_u}^u$ は、ユーザ u が購入したアイテムリストを表す。ここで m_u は、ユーザ u が購入したアイテムの数を表す。 p_{u, r_{j-1}^u, r_j^u} はユーザ u について、直前のアイテムが r_{j-1}^u のとき、次に購入するアイテムが r_j^u となる確率を表す。アイテム r' は、ユーザ u の購買履歴に存在しないアイテムであり、ネガティブアイテムとする。パラメータはシグモイド関数 $\sigma(\hat{p}_{u, r_{j-1}^u, r_j^u} - \hat{p}_{u, r_{j-1}^u, r'})$ の値が最も高くなる値、すなわち p_{u, r_{j-1}^u, r_j^u} が $p_{u, r_{j-1}^u, r'}$ よりも、できるだけ高くなるように更新される。また、ユーザ u がアイテム r_{j-1}^u を購入したとき、左辺のアイテムの確率が右辺のアイテムの確率より高いという大小関係を表す不等号を $>_{u, r_{j-1}^u}$ とする。パラメータ Θ の更新は式 (4) のように定式化される。

$$\begin{aligned} \hat{\Theta} &= \arg \max_{\Theta} \ln \prod_{u \in \mathcal{U}} \prod_{r_j^u \in S^u} \prod_{r' \notin S^u} \Pr(r_j^u >_{u, r_{j-1}^u} r' | \Theta) \Pr(\Theta) \\ &= \arg \max_{\Theta} \sum_{u \in \mathcal{U}} \sum_{r_j^u \in S^u} \sum_{r' \notin S^u} \ln \sigma(\hat{p}_{u, r_{j-1}^u, r_j^u} - \hat{p}_{u, r_{j-1}^u, r'}) \\ &\quad - \Omega(\Theta). \end{aligned} \quad (4)$$

ここで $\Theta = \{\beta_j, \vec{I}_i, \vec{t}_u, \vec{t}' | I_i \in \mathcal{I}, u \in \mathcal{U}\}$ であり、すべてのパラメータの集合である。また、 $\Omega(\Theta)$ は正則化項であり、本論文では L_2 正則化を用いている。

3. 提案手法

3.1. 着想

TransRec ではユーザ u の嗜好ベクトル \vec{T}_u と最後に購入したアイテム r_{j-1}^u のみを使って推薦を行うため、(1) ユーザが連続性のある購買行動を行った際の影響をモデル化できない、(2) ユーザが単発的なイベントで購入した際には、次の購買行動との関係性を表現することが難しい、という2つの課題がある。まず (1) について、ユーザの購買嗜好は過去の複数の購入アイテムに影響を受ける可能性がある。映画を例とすると、あるユーザが複数のシリーズで構成されている映画を途中まで見ていた場合、次の購入ではその映画の次のシリーズが選択される可能性が高い。しかし、TransRec は複数のアイテムの情報を考慮することができないため、ユーザの複数個前からの購入による嗜好への影響をモデル化することが難しい。(2) について、直前のアイテムを突発的な理由で購入した場合、次の購入へユーザの嗜好を適切に反映できない可能性がある。例えば、普段はコメディ映画を好んでみるユーザが、偶然友人達と共にホラー映画を見た場合、TransRec の予測では最後に購入した商品のみをもとに次のアイテムが推薦されてしまう。そのため、推薦リストの中からユーザが本来は興味を持っていないとしても、ホラー映画を推薦してしまう。

そのため、直前のアイテムのみを用いる代わりに、直近 N 個のアイテムを考慮することによりユーザの購買順序を考慮し、ユーザの偶発的な購入の影響を受けにくいモデルを構築する。この場合、ユーザの連続に購入した商品の前後関係を学習することで、ユーザ嗜好の時間推移を正しく推測するこ

とが可能となる。そのようなモデルとして、本研究では、過去 N 個のアイテムの情報を考慮した N -TransRec を提案する。

3.2. 提案手法

提案手法である N -TransRec は TransRec と同様に、全ユーザの嗜好ベクトルと全アイテムの特徴ベクトルを K -次元空間に埋め込み、この空間内でユーザと購入アイテムの関係を学習する。ここで、購入アイテム r_j^u の購入確率は、直近 N 回の購入アイテムに対応したベクトル $\vec{r}_{j-N}^u, \dots, \vec{r}_{j-1}^u$ およびユーザ u の嗜好ベクトル \vec{T}_u の重み付き線形和で与えられるモデルを考える。各アイテムベクトルの重み α_k を用いて、 N -TransRec は遷移関係を式 (5) で定義する。

$$\vec{T}_u + \sum_{k=1}^N \alpha_k \vec{r}_{j-k}^u \approx \vec{r}_j^u. \quad (5)$$

すなわち、購買履歴の直近 N 個のアイテムの購入順序をもとに重要度 α_k を任意に設定することができる。ここで、 \vec{r}_{j-k}^u はユーザ u の購買履歴において、 j 番目のアイテム r_j^u の k 個前の購入アイテム r_{j-k}^u のベクトルを表す。ユーザの嗜好 \vec{T}_u は TransRec と同様に、全ユーザ共通の嗜好 \vec{t} とユーザ個人の嗜好の方向性ベクトル \vec{t}_u から構成され、式 (6) で定義される。

$$\vec{T}_u = \vec{t} + \vec{t}_u. \quad (6)$$

各アイテムの購入確率はそのアイテムの人気度 β_j と式 (5) から得られた予測値との距離によって決まる。ユーザ u が最近購入した N 個のアイテム $S_{j-N}^u, \dots, S_{j-1}^u$ が与えられたとき、次に購買されるアイテム r_j^u は式 (7) で定義される。

$$\begin{aligned} \text{prob}(r_j^u) &\propto \beta_j - d\left(\vec{T}_u + \sum_{k=1}^N \alpha_k \vec{r}_{j-k}^u, \vec{r}_j^u\right), \\ \text{s.t. } \vec{r}_{j-k}^u, \vec{r}_j^u &\in \Psi \subseteq \Phi, \forall r_{j-k}^u, r_j^u \in \mathcal{I}. \end{aligned} \quad (7)$$

例えば、 $N=3$ の場合、そのイメージは図1のようになる。

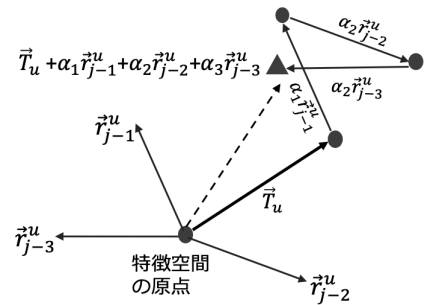


図 1: N -TransRec のイメージ図 ($N=3$)

本研究ではパラメータの更新に際し、Sequential Bayesian Personalized Ranking (S-BPR) を用いて、購買リストから次に購入したアイテム r_j^u を他のすべてのアイテム r' よりも上位にランク付けすることができるように学習を行う。 $preN$ は直前 N 個のアイテム $r_{j-N}^u, \dots, r_{j-1}^u$ を意味し、 $p_{u, preN, r_j^u}$ はユーザ u について、前に購入した N 個のアイ

テムが $r_{j-N}^u, \dots, r_{j-1}^u$ のとき、次に購入するアイテムが r_j^u である確率を表す。アイテム r' はユーザ u の購入履歴に存在しないアイテムであり、ネガティブアイテムと呼ぶ。パラメータはシグモイド関数 $\sigma(\hat{p}_{u,preN,r_j^u} - \hat{p}_{u,preN,r'})$ の値が最も高くなる値、すなわち $p_{u,preN,r_j^u}$ が $p_{u,preN,r'}$ よりも、できるだけ高くなるように更新される。また、 $>_{u,preN}$ をユーザ u が直近に N 個のアイテム $r_{j-N}^u, \dots, r_{j-1}^u$ を購入した上で左辺のアイテムの確率が右辺のアイテムの確率より高いような大小関係を表す不等号とする。パラメータ $\Theta = \{\beta_j, \vec{I}_j, \vec{t}_u, \vec{t} \mid I_j \in \mathcal{I}, u \in \mathcal{U}\}$ の更新は式 (8) のように定式化される。

$$\begin{aligned} \hat{\Theta} &= \arg \max_{\Theta} \ln \prod_{u \in \mathcal{U}} \prod_{r_j^u \in S^u} \prod_{r' \notin S^u} \Pr(r_j^u >_{u,preN} r' \mid \Theta) \Pr(\Theta) \\ &= \arg \max_{\Theta} \sum_{u \in \mathcal{U}} \sum_{r_j^u \in S^u} \sum_{r' \notin S^u} \ln \sigma(\hat{p}_{u,preN,r_j^u} - \hat{p}_{u,preN,r'}) \\ &\quad - \Omega(\Theta). \end{aligned} \quad (8)$$

ここで、 $\Omega(\Theta)$ は正則化項であり、本論文では L_2 正則化を用いる。

4. 実験

本章では、Amazon レビューデータを用いて、従来手法と提案手法の予測精度について検証する。このとき、ハイパーパラメーター N と重み α_k を変化させて、その影響について考察を行う。

4.1. 対象データ

実験では公開された 1996 年から 2014 年の Amazon レビューデータセットのうち、最新の 2 年間である 2013 年から 2014 年の「映画」データセットを用いた [4]。なお、映画鑑賞は一般にシリーズ性があり、強い連続性があると考えられるため、「映画」データセットを用いて実験を行った。このレビューデータを購入履歴データとみなして実験を行う。ここで、直近の 10 個のアイテムを考慮し実験を行うため、前処理として評価数が 12 未満のユーザや評価数が 12 未満のアイテムのデータを削除した。そのため、本実験で用いたデータセットのユーザ数は 10,643 人、アイテム数は 10,833 となった。

4.2. 評価指標

対象となるユーザが最後に購入したアイテム $S_{m_u}^u$ をテストに使用し、残りのすべての $S_1^u, S_2^u, \dots, S_{m_u-1}^u$ を学習データとして使用する。すべてのユーザについてパラメータの学習を行う。 N -TransRec と TransRec の性能を評価するため、式 (9), (10) に示した C_R と C_{H50} の 2 つの指標を用いる。

$$C_R = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \frac{1}{|\mathcal{I} \setminus S^u|} \sum_{r' \in \mathcal{I} \setminus S^u} \mathbf{1}(R_{u,g_u} < R_{u,r'}), \quad (9)$$

$$C_{H50} = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \mathbf{1}(R_{u,g_u} \leq 50). \quad (10)$$

ここで、 g_u は、ユーザ u が実際に購入した次のアイテムを表し、 R_{u,g_u} はユーザ u のアイテム g_u の予測購入確率の順位を表す。 $\mathbf{1}(\mathbf{A})$ は、事象 \mathbf{A} が真であれば 1、そうでな

ければ 0 を返すインジケータ関数である。 C_R は、実際に購入したアイテムのランキングが、全てのアイテムのランキングの中で高いほど大きな値をとる評価指標である。 C_{H50} は、予測された購買確率の高いアイテムが推薦リストにおいて、上位 50 位以内に入っているテストデータの数が多いほど大きな値をとる指標である。いずれも、評価指標の値が大きいほど性能が良いことを示す。

4.3. 実験条件と設定

低次元な特徴空間での説明性と解釈性を確保するために、特徴空間の次元数 K は 10 とし、正則化パラメータ λ については事前実験を行い最も高い精度を示した値を用いた。 N 個のアイテムベクトルに与えられる重み α_k は、 $\alpha_k = 1$ 、 $\alpha_k = 1/N$ 、と $\alpha_k = 1/k$ の 3 通りに設定した。 $\alpha_k = 1$ の場合、直近 N 個のアイテム全てに同じ重要度を与えることと等価である。また、 $\alpha_k = 1/N$ の場合、直近 N 個のアイテムのそれぞれの重みは同じ値であるが $1/N$ 倍されるため、 N が大きくなるにつれて各アイテムの重さが平等に小さくなり、 $\alpha_k = 1$ の場合よりユーザの嗜好ベクトル \vec{t}_u の影響を大きくした設定である。最後に $\alpha_k = 1/k$ の設定、最近購入されたアイテムの方が、それ以前に購入されたアイテムに比べその影響が大きくなるようなケースである。

4.4. 実験結果

実験結果を表 1～表 3 に示す。

表 1: 学習データに対する C_R

	$\alpha_k = 1$	$\alpha_k = 1/N$	$\alpha_k = 1/k$
従来	0.857	0.857	0.857
提案 ($N=2$)	0.906	0.836	0.887
提案 ($N=3$)	0.920	0.821	0.897
提案 ($N=4$)	0.925	0.816	0.900
提案 ($N=5$)	0.930	0.807	0.900
提案 ($N=6$)	0.931	0.804	0.897
提案 ($N=7$)	0.934	0.807	0.895
提案 ($N=8$)	0.936	0.809	0.892
提案 ($N=9$)	0.938	0.811	0.891
提案 ($N=10$)	0.940	0.812	0.888

表 2: テストデータに対する C_R

	$\alpha_k = 1$	$\alpha_k = 1/N$	$\alpha_k = 1/k$
従来	0.820	0.820	0.820
提案 ($N=2$)	0.848	0.816	0.842
提案 ($N=3$)	0.859	0.808	0.850
提案 ($N=4$)	0.860	0.803	0.852
提案 ($N=5$)	0.861	0.796	0.850
提案 ($N=6$)	0.863	0.793	0.848
提案 ($N=7$)	0.861	0.790	0.846
提案 ($N=8$)	0.857	0.790	0.842
提案 ($N=9$)	0.852	0.790	0.836
提案 ($N=10$)	0.850	0.786	0.831

この結果から、 N -TransRec は従来の手法に比べて 2 つの評価指標のいずれの場合でも精度が高いことが示された。さらに t 検定を行い、すべての精度向上が有意水準 1% で統計的に有意であることを確認した。 $\alpha_k = 1$ の時、学習データにおける C_R は N の増加により継続的に精度が向上する傾向を示したが、テストデータにおける C_R は $N = 6$ の場合 ($N = 6, \alpha_k = 1$) に、最も高い値を達成した。 C_{H50} は N -TransRec モデル ($N = 7, \alpha_k = 1$) が最も良いことを示した。これらの結果は、TransRec に基づく N 回前まで

表 3: テストデータに対する C_{H50}

	$\alpha_k = 1$	$\alpha_k = 1/N$	$\alpha_k = 1/k$
従来	0.141	0.141	0.141
提案 ($N=2$)	0.173	0.132	0.149
提案 ($N=3$)	0.177	0.124	0.147
提案 ($N=4$)	0.178	0.117	0.156
提案 ($N=5$)	0.174	0.127	0.154
提案 ($N=6$)	0.174	0.129	0.154
提案 ($N=7$)	0.185	0.131	0.158
提案 ($N=8$)	0.175	0.117	0.153
提案 ($N=9$)	0.167	0.126	0.144
提案 ($N=10$)	0.164	0.130	0.150

の商品購買の順序情報を考慮した推薦システムが有効であることを示唆している。

5. 実データ分析

対象データでの人気映画を対象に、類似度が高い 10 個の映画を抽出して分析を行う。一例として Movie データセットにおける人気商品「Star Trek Into Darkness」との類似度が高い 10 個の商品を表 4 に、それぞれ所属するジャンルを表 5 に示す。

表 4: 人気商品との類似度 Top10 の商品

順位	タイトル
1	Pacific Rim
2	Beautiful Creatures
3	I, Robot
4	The Great Discovery
5	Crime Spree
6	Prometheus
7	Ender's Game
8	The Chronicles of Narnia III
9	Premium Rush
10	World War Z

表 5: 人気商品との類似度 Top10 商品が所属するジャンル

順位	アクション	SF 映画	アドベンチャー
1	✓	✓	✓
2	✗	✓	✗
3	✓	✓	✗
4	✗	✗	✗
5	✓	✗	✗
6	✓	✓	✓
7	✓	✓	✓
8	✗	✓	✓
9	✓	✗	✗
10	✓	✓	✓

ここで、✓ はその順位の映画が対応するジャンルに所属することを意味し、✗ は所属しないことを表す。「Star Trek Into Darkness」はアクション、SF 映画、アドベンチャーに所属している。得られた埋め込み表現が適切であったかどうかについてジャンル情報を用いて検証したところ、9 割ほど同じジャンルのアイテムが存在しており、得られた特徴空間では類似したアイテムの距離が近づくように学習できていることがわかる。以上より、嗜好が反映された埋め込み表現

が得られたと考えられる。

6. 考察

本研究では、複数の商品購買順序情報を考慮する N -TransRec モデルを提案した。さらに購買順序によって各アイテムベクトルの重さ α_k を変化させ、 $\alpha_k = 1$ の場合が最も優れていることを明らかにした。 $\alpha_k = 1$ は、直近 N 個の購入アイテムに対応したベクトル $r_{j-N}^u, \dots, r_{j-1}^u$ およびユーザ u の嗜好ベクトルの和で表現し、購買順序に関わらず前の N 個のアイテム全てに同じ重視度を与えることを示している。さらに $\alpha_k = 1$ の条件下では、 $N = 7$ のときにテストデータにおける C_{H50} が一番高い値を示した。これは、直前の一点ではなく、購買履歴から直前 7 点の購入商品を平等に加味したとき、次の商品の予測において最もよい精度を示したことを意味している。ただし予測時には、直前の 7 個のアイテムベクトルが大きな割合を占めるようになり、ユーザの嗜好ベクトル T_u の重要度が減少する点に留意する必要がある。さらに提案手法では、重み α_k と N の値を変換し、様々なモデルを構築することが可能であり、頑健性があると考えられる。

最後に、実応用について考える。実世界では、購買の行動やそれらを取り巻く環境は多様で複雑である。しかし、 N -TransRec は、ユーザとアイテムの購買リストから複数の購入商品の情報を扱い、複雑な内部関係を捉えて推薦を行うことができるため、実応用での有効性が期待できる。

7. まとめと今後の課題

本論文では、埋め込み空間内でユーザとアイテムの関係を学習する TransRec モデルを拡張し、連続したユーザの購買履歴を考慮した埋め込みモデルである N -TransRec を提案した。 N -TransRec は最近の N 個のアイテムを考慮することで、ユーザ嗜好ベクトルとアイテムベクトルを再構築し、情報をより多く含めて推薦を行うモデルである。

公開データセットを用いた実験を行った結果、提案手法の精度は従来手法より優れていることがわかった。今後の課題として、ユーザの嗜好の変化を活用し N 個のアイテムの重み α_k の適切な設定による、アイテムの推薦精度の向上が挙げられる。

参考文献

- [1] Ruining He, Wang-Cheng Kang, and Julian McAuley. Translation-based recommendation. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pp. 161–169, 2017.
- [2] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. *ArXiv Preprint ArXiv:1310.4546*, 2013.
- [3] Oren Barkan and Noam Koenigstein. Item2vec: neural item embedding for collaborative filtering. In *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6. IEEE, 2016.
- [4] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 43–52, 2015.