

Attention 機構の導入と単層強化手法による NGCF モデルの改善策に関する検討

陳書婷¹ 高橋聡² 北澤正樹^{3, 1} 吉川厚^{4, 1}

概要: ニューラルグラフ協調フィルタリング (NGCF) モデルは、近年推薦システムで広く利用されている。本研究では、ユーザーやアイテムといった各ノードは自身の接続状況によって異なる重要度分布を持つと考えられることから、NGCF モデルにおいて各層の埋め込みが同じ重みを持つことを課題と捉え、積層問題として指摘した。そして、この積層問題を解決するために、Attention 機構の導入と単層強化の2つの改善策を提案した。Attention 機構の導入はノードの層間重みの組み合わせを自動的に学習する手法であり、単層強化はモデルの各層を順番に1層のみ増強して他の層を変更しないようにする手法である。それぞれの手法において、2つのデータセットを用いて従来研究との比較実験を行った結果、推薦精度が向上した。本研究の提案手法は積層問題の解決に効果的であると考えられる。

キーワード: 推薦システム, グラフニューラルネットワーク, 協調フィルタリング, Attention 機構

A Study on the Improvement of NGCF Model by Introducing Attention Mechanism and Single Layer Reinforcement Method

SHUTING CHEN¹ SATOSHI TAKAHASHI²
MASAKI KITAZAWA^{3,1} ATSUSHI YOSHIKAWA^{4,1}

Abstract: Neural Graph Collaborative Filtering (NGCF) model has been widely used in recommendation systems in recent years. In this study, we pointed out that each node of users and items should have different importance distributions according to its own connection status, and it is not appropriate for each layer of embedding to have the same weight in the NGCF model, which we call the stacking problem. To solve this stacking problem, we proposed two improvements: the introduction of an attention mechanism and single-layer reinforcement. the introduction of an attention mechanism is a method that automatically learns the combination of weights between layers of nodes, and single-layer reinforcement is a method that augments each layer of the model in turn by only one layer and leaves the other layers unchanged. This study was conducted on two datasets. We conducted comparative experiments on two datasets and showed that the proposed method can improve the recommendation accuracy. We verified that the proposed method helped to solve the stacking problem.

Keywords: Recommendation system, Graph neural network, Collaborative filtering, Attention network

1. はじめに

1.1 研究背景

情報爆発の時代において、推薦システムは情報過多を緩和するために極めて重要な役割を果たし、多くのオンラインサービスで広く採用されている。個人化推薦システムの鍵となるのは、ユーザーの過去のインタラクション（評価やクリックなど）に基づいてアイテムに対するユーザーの嗜好をモデリングすることであり、これは協調フィルタリング (Collaborative Filtering, CF) として知られている[1,2]。CF 手法の共通のパラダイムは、ユーザーとアイテムの歴史的な相互作用を通して、ユーザーとアイテムの表現（別名：エンベディング）を学習することである。行列分解 (Matrix Factorization, MF) [3,4]に代表される初期の協調フィルタリング手法では、潜在的な特徴のベクトルを用いてユーザーやアイテムを表現し、ユーザーやアイテムを共有

の潜在空間に投影する。そして、アイテムとユーザーのインタラクションを、両者の潜在的なベクトルの内積としてモデル化するものである。これまでの MF ベースのアプローチでは、ユーザーとアイテムのインタラクショングラフに存在する潜在的な関係性を無視していた。その後の研究で、ユーザーのインタラクション履歴をユーザー表現にさらに組み込むことで、推薦結果を改善できることがわかった[4,5]。これらの改善は、推薦システムにおいてユーザーのワンホップネイバーをモデル化することの利点を示している。しかし、前述の推薦モデルでは、マルチホップネイバーの暗黙の協調情報を考慮されていない。

近年では、ノード情報とトポロジーを自然に統合できる GNN (Graph Neural Networks, GNN) 技術が注目されている[6]。GNN は、ユーザーとアイテムのインタラクションに関する情報を取得し、ユーザーとアイテムの埋め込み表現

¹ 立教大学 大学院人工知能科学研究科
Graduate School of Artificial Intelligence and Science, Rikkyo University
² 関東学院大学 理工学部
College of Science and Engineering, Kanto Gakuin University

³ 北澤技研
Kitazawa Tech
⁴ 東京工業大学 情報理工学院
School of Computing, Tokyo Institute of Technology

を学習することができるため、GNN をレコメンドシステムに利用する分野がブームになっている。より古典的なモデルの 1 つに Wang らが提案したニューラルグラフ協調フィルタリング (Neural Graph Collaborative Filtering, NGCF) がある[7]。これは、グラフ畳み込みネットワーク (Graph Convolutional Networks, GCN) を推薦タスクに適用し、有望な結果を得ている。ユーザーとアイテムのインタラクショングラフの各ノードは、特定のセマンティクスを持たない ID を入力としているだけである[8]。

NGCF モデルが提案された後、情報集約プロセスの改善と補助的な情報の取り込みという 2 つの方法でモデルを改善するための研究が数多く行われてきた[8,9,10,11]。これらのモデルは、情報集約プロセスの改善や補助的な情報の取り込むことにより、いずれもうまく機能するようになった。しかし、いずれも最終的な埋め込みを形成するために各層の埋め込みを接続する際に、層間重要度を区別していない。これらの研究では、各ノード自体の集約層間での重要度分布を無視したものとなっている。本研究では、この問題を「積層問題」と呼ぶ。

1.2 積層問題

本研究における「積層問題」とは、各ノードの最終的な埋め込みは、各畳み込み層が等しい重みで重ね合わせ、ノードごとに異なる集計層は異なる重要度が区別されていないことである。

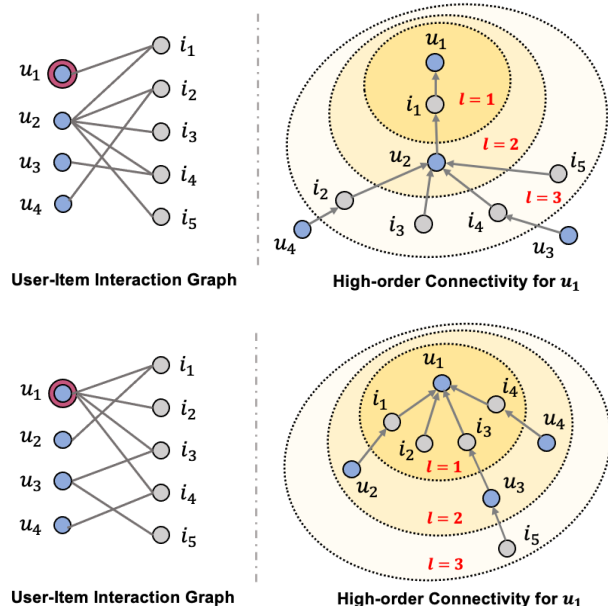


図 1 異なる接続状態での u_1

図 1 には、2 つの異なる接続シナリオでのノード u_1 を示している。左の u_1 は、各層でほぼ同じ数の隣接ノードを持っている。一方、右の u_1 は、層 1 と層 2 ではほとんど隣接ノードを持っていないが、層 3 では多くのノード情報を持っていることがわかる。つまり、右の u_1 にとっての 3 層目の重要度が左の u_1 にとっての重要度より高い。そこで、各

ノードが自身の接続状況によって異なる重要度分布を持つべきであり、NGCF モデルでは各層の埋め込みが同じ重みを持つことは適切ではないと考えられる。

K 層 GNN の各ノードは、ノード自身とその K ホップネイバーを含むノードのセットを取り込む。このセットを Receptive Field (RF) と呼ぶ[12]。現在の推薦アルゴリズムでは、特徴の伝搬に固定層を採用しているため、すべてのノードの RF が固定されてしまう。このようなグラフの伝搬では、異なる受信フィールドの下でノード特徴の興味深い相関関係をモデル化する柔軟性に欠ける。これにより、RF のサイズが小さいために長距離依存性を十分に活用できなかったり、RF のサイズが大きいために過度に平滑化されたノイズが導入されて局所的な情報が失われたりする。どちらの場合も、識別性の高いノード表現としては最適ではない。

最近では、重みの学習による GCN の改良に関する研究も盛んに行われている。Song らは、隣人を意識した新しい Graph Attention 層を採用している。これは、あるノードの異なる隣人に対して、その隣人間の類似度をペアで計算することにより、異なる隣人を意識した Attention 係数を割り当てるものである[8]。Song らは、各ノードの隣人間の暗黙の相関を考慮し、隣人間の相関関係が集約係数に与える影響を検証したが、層間の重みの分布は考慮していない。しかし、これらの手法は、その有効性にもかかわらず、ノードからの特徴の組み合わせは考慮していない[13]。

1.3 研究目的

本研究では、NGCF モデルのノードの各層の重みを変更することで、NGCF の積層問題を解決し、最終的により良い推薦性能を得ることを目的とする。NGCF モデルでは、各層の埋め込みと初期埋め込みが等しい重みで連結され、最終的な埋め込みを得られる。ユーザーとアイテムのインタラクショングラフに各ノードの近隣接続状況が異なるため、本研究では各ノードの接続状況に基づいて層間の重要度を区別することで、モデルの推薦精度を向上させることができると考える。

NGCF モデルの改善方法として、Attention 機構の導入と単層強化の 2 つの手法を提案する。提案手法 1 では、NGCF モデルに Attention 機構を導入し、各ノードの異なる畳み込み層に対応する重要度係数を自動的に学習することを実現する。提案手法 2 は、他の層を変更せず、モデルの各層を順番に強化することである。重要度の高い層の役割を強化することを期待し、モデルの複雑さを増すことなく、単一層に対する強化操作を提案するものである。

本研究では、公開されている 2 つのデータセットを使って実験を行い、提案手法の有効性を評価する。

2. 提案手法 I—Attention 機構の導入

提案手法 1 では, GAMLP にヒント[12]を得て, NGCF モデルにも Attention 機構を導入し, Attention 機構が自動的に重みを割り当てる機能を活用し, 積層問題の解決を試みる.

提案手法 1 のアーキテクチャを図 2 に示す. このアーキテクチャは NGCF モデルをベースに, 各層の埋め込みに適当な重みをつけるように, Attention Network 層を加えたものである. 改良モデルには 4 つのレイヤーがある. (1) Embedding 層: ユーザーエンベッディングとアイテムエンベッディングを提供し初期化する. (2) Embedding 伝搬層: 高次の接続関係を注入してエンベッディングを改良する. (3) Attention Network 層: 各層のノードからの入力を得て, 対応する最適な重みを出力する. (4) 予測層: 各層からのエンベッディングを連結し, 内積してユーザーとアイテムのペアの好みスコアを出力する. 本章で提案モデルの各レイヤーを別々に説明し, 最後に損失関数とデータトレーニングの設定について説明する.

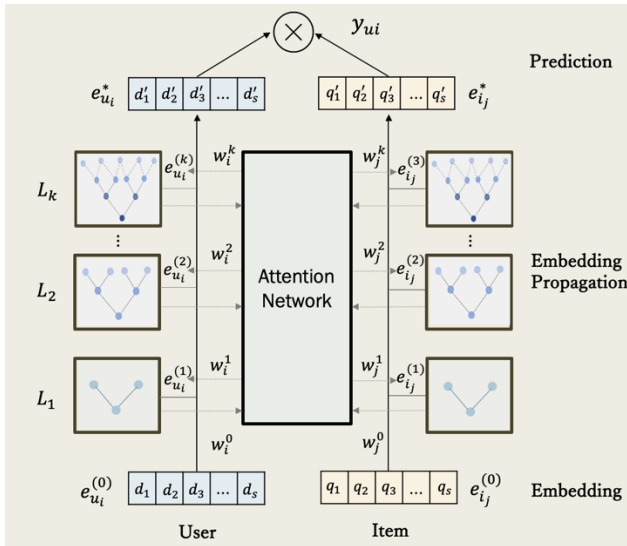


図 2 Attention 機構を導入した NGCF モデル

2.1 Embedding 層

提案アーキテクチャの Embedding 層では, ユーザー u (アイテム i) を, 埋め込みサイズを d とした埋め込みベクトル $e_u \in R^d$ ($e_i \in R^d$) で記述し, ユーザー u の埋め込み e_u とアイテム i の埋め込み e_i を直列して最初の埋め込み E を得る (式 1). これは, パラメータ行列をエンベッディングのルックアップテーブルとして構築していると考えられる. ここで, n はユーザーの数であり, m はアイテムの数である.

$$E = [e_{u_1}, \dots, e_{u_n}, e_{i_1}, \dots, e_{i_m}] \quad (1)$$

このエンベッディングテーブルは, 主流のレコメンダーモデル[14,15,16]と NGCF に従い, ユーザーエンベッディングとアイテムエンベッディングの初期状態として機能し, エンド・ツー・エンドで最適化される. MF や NCF (Neural

Collaborative Filtering) [15]のような伝統的なレコメンダーモデルでは, ユーザーとアイテム ID のエンベッディングは, 予測スコアを達成するためにオペレーター直接与えられる. 対照的に, NGCF フレームワークでは, ユーザーとアイテムのインタラクショングラフ上に伝搬させることで, エンベッディングを洗練させる. 協調的な信号を埋め込みに注入するため, より良い推薦性能が得られる[7].

2.2 Embedding 伝搬層

Embedding 伝搬層では, NGCF モデルと同じ伝搬規則に従って, 特徴変換, 近傍集約, 非線形活性化という形で埋め込みを行う (式 2). LightGCN[9]では, GCN から受け継いだ特徴変換と非線形活性化という 2 つの演算が NGCF の有効性に寄与していないことを示した. さらに, この 2 つの操作を取り除いたことで, 精度を大幅に向上させた. しかし, 本研究の焦点は伝搬規則ではないので, 比較のために, 依然として NGCF と同じ層別伝搬規則に従っている.

$$E^{(l)} = \text{LeakyReLU} \left((\mathcal{L} + I)E^{(l-1)}W_1^{(l)} + \mathcal{L}E^{(l-1)} \odot E^{(l-1)}W_2^{(l)} \right) \quad (2)$$

ここで, $E^{(l)} \in \mathbb{R}^{(n+m) \times d_l}$ は, l 層の埋め込み伝搬の後に得られるユーザーとアイテムの表現である. $E^{(0)}$ は, 初期のメッセージパッシングにおける E として設定される. すなわち, $e_u^{(0)} = e_u, e_i^{(0)} = e_i$ となる. I は恒等行列を表す. $\mathcal{L}$ はユーザー・アイテム・グラフのラプラシアン行列を表し, 次のように定式化される (式 3).

$$\mathcal{L} = D^{-\frac{1}{2}} A D^{-\frac{1}{2}} \quad (3)$$

ここで, A は隣接行列, D は対角次数行列を表し, t 番目の対角要素 $D_{tt} = |N_t|$ となり, 非対角要素 $\mathcal{L}_{ui} = 1/\sqrt{|N_u||N_i|}$ となる.

2.3 Attention Network 層

Attention Network の本質は, 人間の視覚的注意メカニズムに由来している. 人は何かを見るとき, 最初から最後まですべてを見るのではなく, 必要に応じて特定の部分だけを観察し, 注意を払う傾向がある. 簡単に言えば, モデルが重要な情報を捉えやすくすることを目的とした, 重みのあるパラメータを割り当てる仕組みのことである[17]. スケーラブルでノードに適応したグラフ学習をサポートするために, Zhang らは, GAMLP (Graph Attention Multi-Layer Perceptron) と呼ばれる 3 つの RF Attention を持つ新しい MLP (Multi-Layer Perceptron) を提案した[12].

手法 1 では GAMLP の Attention 機構の一つである JK Attention を改良し, NGCF の積層問題を解決するに適用できるようにしたものである. 図 3 は, Attention Network 層の構造を示している. 提案の Attention Network では, 伝搬層の各層のエンベッディングを Attention ネットワークに入力し, 第 1 層から第 l 層までのエンベッディングをまず直列し, 非線形変換して比較ベクトルを得る. そして, す

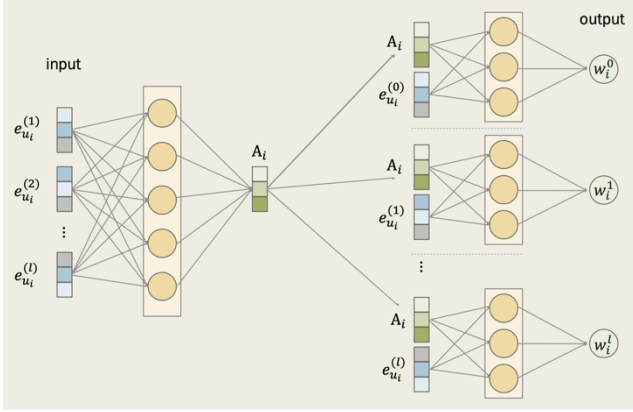


図 3 Attention Network の仕組み

すべての層のエンベディングと比較ベクトルを別々に連結し、非線形変換を行うことで類似度を求める (式 4, 5)。

$$A_i = \tanh \left[\left(e_{u_i}^{(1)} \parallel e_{u_i}^{(2)} \parallel \dots \parallel e_{u_i}^{(K)} \right) W_3 + B \right] \quad (4)$$

$$\tilde{w}_i^l = \tanh \left[\left(e_{u_i}^{(l)} \parallel A_i \right) W_4 \right] \quad (5)$$

トレーニングを続けるうちに、Attention に基づく重み付け分布は、ターゲットノードにとってより有用な近隣領域を徐々に強調するようになる。Attention ネットワークは、各ノードの特徴表現をより良いものにすることができる。

2.4 予測層

予測層では、L 層を伝搬した後、各層の埋め込みを Attention Network 層で学習した注目度の重みと組み合わせ、ユーザーの複数の表現 $\{e_u^{(0)}, e_u^{(1)}, \dots, e_u^{(l)}\}$ 、アイテムの複数の

の表現 $\{e_i^{(0)}, e_i^{(1)}, \dots, e_i^{(l)}\}$ が得られる。それらを直列し、ユーザーとアイテムの最終埋め込みを求める (式 6)。

$$e_u^* = e_u^{(0)} \parallel \dots \parallel e_u^{(l)}, \quad e_i^* = e_i^{(0)} \parallel \dots \parallel e_i^{(l)} \quad (6)$$

最後に、ターゲットアイテムに対するユーザーの嗜好を推定するための内積を行う (式 7)。

$$\hat{y}(u, i) = e_u^{*T} e_i^* \quad (7)$$

2.5 損失関数

提案アーキテクチャの損失関数は、ペアごとの Bayesian Personalized Ranking 損失[16]を最適化するものであり、推薦システムで広く使用されている[17,18]。これは、観察されたユーザーアイテムの相互作用と、観察されていないユーザーアイテムの相互作用の間の相対的な順序を考慮するものである。具体的には、ユーザーの嗜好をより反映している観察されたインタラクションは、観察されていないインタラクションよりも高い予測値が割り当てられるべきであるとしている。目的関数は、以下のように設定する。

$$Loss = \sum_{(u,i,j) \in \mathcal{O}} -\ln \sigma(y_{ui} - y_{uj}) + \lambda \|\theta\|_2^2 \quad (8)$$

ここで、 $\mathcal{O} = \{(u, i, j) | (u, i) \in \mathcal{R}^+, (u, j) \in \mathcal{R}^-\}$ はペアごとの学習データ、 $\mathcal{R}^+$ は観測された相互作用、 $\mathcal{R}^-$ は観測されていない相互作用、 $\sigma(\cdot)$ はシグモイド関数、 $\theta = \{E, \{W_1^{(l)} W_2^{(l)}\}_{l=1}^L\}$ は学習可能なすべてのモデルパラメータ、 λ はオーバーフィッティングを防ぐための L2 正則化の強さを制御する。

3. 提案手法 II—単層強化

3.1 アーキテクチャ

Attention Network の複雑なアーキテクチャとは対照的に、本研究はよりシンプルで効果的なアプローチである単層強化手法を提案した。

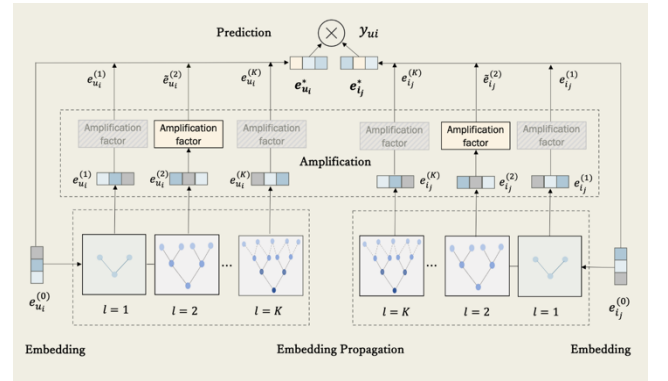


図 4 提案手法 2 のアーキテクチャ。この図では全ノードに対して 2 層目の埋め込みに増幅率をかけて増強することを示している。

単層強化手法は、ある層を選択し、他の層の埋め込みはそのままに、選択した層を強化することで推薦性能の向上を求めるというものである。この手法では、各ノードが層ごとに異なる重みを持つことに注目するのではなく、総層数 K である NGCF モデル全体を見て、どの層が最も貢献度が高いかを議論している。本研究では、最も貢献度の高い層を強化することで、モデルの精度を向上できると考えている。そこで、モデルの学習パラメータを増やさず、訓練をシンプルにするために、モデルの各層を順番に 1 層ずつ増強し、他の層を変更しないようにした。この手法では、いくつかの固定増幅率 z が選ばれ、増強層のすべてのノードの埋め込みを強化する。各層の強化した後に最も効果的な結果を選択する。この手法のアーキテクチャを図 4 に示す。

提案モデルには 4 つのレイヤーがある。(1) Embedding 層、(2) Embedding 伝搬層、(3) Amplification 層、(4) 予測層である。提案手法 2 のうち、Embedding 層、Embedding 伝搬層、予測層は提案手法 1 と同様である。そこで、本章では新たに追加する Amplification 層についてのみ説明する。

3.2 Amplification 層

Amplification 層では、伝搬後の各埋め込み層のうちに f 番目の層を強化対象として選択し、その層の全ノードの埋め込みに増幅率 z を乗算する。そして、乗算の結果を第 f 層の新しい埋め込みとする。その際、他の層の埋め込みは変更しないようにする (式 9,10)。

$$\tilde{e}_u^{(f)} = ze_u^{(f)}, \tilde{e}_i^{(f)} = ze_i^{(f)} \quad (9)$$

$$\tilde{e}_u^{(l)} = e_u^{(l)}, \tilde{e}_i^{(l)} = e_i^{(l)} \quad (l \in \{1, 2, \dots, K | l \neq f\}) \quad (10)$$

ここで、 l は l 番目の層、 K は層の総数を表す。強化対象層 f は手作業で選ぶため、 f の範囲は $\{1, 2, \dots, K\}$ で選択する。増幅率 z は $\{1.5, 2.0, 2.5, 3.0\}$ で調整される。

4. 実験

4.1 データセット

提案手法の有効性を評価するため、2 つのベンチマークデータセットで実験を行う。Gowalla[a]と Movielens 100K[b]である。この 2 つのデータセットは一般に公開されており、ドメイン、サイズ、スパース性の点で異なっている。この 2 つのデータセットの統計情報を表 1 にまとめる。

表 1 データセットの統計情報

Datasets	#Users	#Items	#Interactions	Density
Movielens	943	1,682	100,000	0.06305
Gowalla	29,858	40,981	1,027,370	0.00084

4.2 実験設定

4.2.1 評価指標

本研究ではすべての実験において、比較対象として Wang らの NGCF モデル[7]を用いる。NGCF モデルと比較するために、従来研究と同様に、広く使われている 2 つの評価指標、 recall@Q と ndcg@Q を採用した[8]。どちらの指標でも従来研究と同様に $Q=20$ と設定した。テストセットの全ユーザーの平均メトリックを報告する。

4.2.2 パラメータ設定

モデルを Tensorflow で実装している。埋め込みサイズはすべてのモデルで 64 に固定されている。バッチサイズを 1024 に固定し、Adam オプティマイザでモデルを最適化する。ハイパーパラメータについては、学習率を $\{0.0001, 0.0005, 0.001, 0.005\}$ 、L2 正則化係数を $\{10^{-5}, 10^{-4}, \dots, 10^1, 10^2\}$ で調整し、ドロップアウト比率を $\{0.0, 0.1, \dots, 0.8\}$ の中からグリッドサーチを行っている。さらに、検証データの recall@20 が 50 回連続して増加しない場合には早期に停止するという早期停止戦略を行っている。

4.3 提案手法 I

提案手法 1 の実験では、2 つのシナリオで実験を行った。

実験 1 は、すべてのノードが各層で共通の層間重み分布を持ち、異なる畳み込み層では異なる重みを持つ。実験 2 は、各ノードが各層で独自の異なる重みを持つ。両実験の重みは、Attention Network を使って学習する。これは、徐々にノードが独立化していくプロセスとも言える。

4.3.1 実験 1

すべてのノードが各層で共通の層間重み分布を持つ場合の実験結果を表 2 に示す。Movielens と Gowalla の両データセットにおいて、我々の提案する改良モデルは、元のモデルと比較して改善されていることがわかる。Attention ネットワークでは、各層のノードの重みを学習させていないが、単純に各層の重みを総合的に区別することで、すでにある程度の改善が見られる。

表 2 実験 1 の結果

	Movielens		Gowalla	
実験 1	recall	ndcg	recall	ndcg
NGCF	0.04873	0.10052	0.15567	0.22371
改良モデル	0.05013	0.10189	0.15636	0.22780
%Improve	2.87%	1.36%	0.44%	1.83%

4.3.2 実験 2

各ノードが各層で独自の異なる重みを持つ場合の実験結果を表 3 に示す。見ての通り、改良モデルは Movielens データセットで良好な性能を発揮し、システムの推薦効果を高めている。しかし、Gowalla データセットでは、その効果はそれほど明らかではない。

表 3 実験 2 の結果

	Movielens		Gowalla	
実験 1	recall	ndcg	recall	ndcg
NGCF	0.04873	0.10052	0.15567	0.22371
改良モデル	0.05149	0.10168	0.15363	0.21819
%Improve	5.67%	1.16%	-1.31%	-2.47%

4.3.3 考察

実験 1 と実験 2 では、ノードが各層で異なる重みを持つことができる Attention 機構を導入することで、推薦システムの性能をある程度向上させることができることを検証した。また、実験 2 で Gowalla が期待した効果を得られなかった原因として、主に二つの要因が想定される。

第一に、Gowalla は非常に大きなデータセットであるため、実験では学習に時間がかかり、モデルの収束も非常に遅いことがわかった。前節で述べたように、NGCF における特徴変換と非線形活性化は、協調フィルタリングに悪影響を与えている[9]。Attention Network の追加により、元の

a) <http://snap.stanford.edu/data/loc-gowalla.html>

b) <https://grouplens.org/datasets/movielens/100k/>

モデルがさらに複雑になり、トレーニングが困難になる。したがって、顕著な改善が見られなかった一因は、データセット自体のサイズが大きいため、冗長な伝搬規則を持つモデルでは Attention 機構がうまく機能しなかったためと考えている。

第二に、ノードの各層の埋め込みはそれ自体がパラメータベクトルであるため、埋め込みを訓練する同時にその埋め込みの重みを訓練すると、学習目的が不明瞭になる可能性がある。その結果、良い埋め込み表現も適切な重みも学習できず、モデルの有効性が低下してしまうと考えている。そのため、多くの深層学習に関連するモデルでは、モデルの構成要素を別々に学習させるようになっているのである。本研究をベースに、Attention 機構を個別にトレーニングすることも、今後検討したい課題である。

4.4 提案手法II

4.4.1 性能比較

提案手法 2 の実験では、増幅率 z を[1.5, 2.0, 2.5, 3.0]の順で設定した。そして、異なる z の値のうえで、総層数 K が 2, 3, 4 である 3 つのモデルで別々に実験を行う。各モデルの実験で、1 層目から順に強化対象として選択し、すなわち、強化する層 f の値を[1, 2, ..., K]の順で設定した。最後に、モデルの各層の強化結果を統計し、強化しない NGCF モデルの基準値と比較し、最大改善率を得る。表 4 は、増幅率 $z = 1.5$ の場合に各層を強化した結果のまとめである。表 5 は、増幅率 $z = 2.0$ の場合に各層を強化した結果のまとめである。表 6 は、増幅率 $z = 2.5$ の場合の結果のまとめである。表 7 は、増幅率 $z = 3.0$ の場合の結果のまとめである。

表 4 増幅率 $z = 1.5$ の場合に各層を単層増強した結果

	Movielens		Gowalla	
	recall	ndcg	recall	ndcg
K=2				
NGCF	0.04807	0.09665	0.15515	0.21447
強化層 $f=1$	0.05065	0.10075	0.15918	0.22997
強化層 $f=2$	0.05039	0.09912	0.15906	0.22825
%Improve(max)	5.37%	4.24%	2.60%	7.23%
K=3				
強化層 $f=1$	0.05471	0.10354	0.16046	0.23122
強化層 $f=2$	0.05381	0.10819	0.15912	0.22809
強化層 $f=3$	0.04666	0.09787	0.15692	0.22537
%Improve(max)	12.27%	7.6%	3.08%	3.36%
K=4				
強化層 $f=1$	0.05160	0.10154	0.15917	0.22846
強化層 $f=2$	0.05151	0.09833	0.15587	0.22438
強化層 $f=3$	0.04903	0.09609	0.15626	0.22619
強化層 $f=4$	0.04775	0.09508	0.15512	0.22465
%Improve(max)	2.42%	1.09%	3.02%	2.08%

4.4.2 考察

表 4 の実験結果により、増幅率 $z = 1.5$ の場合、両方のデータセットにおいてモデルの単層増強を行った後、評価指標に大きな改善が見られる。特に Movielens は、総層数 $K = 3$ の場合、第 1 層の強化で 12.27%の改善を達成している。さらに、第 1 層の強化効果が最も明らかなで、すべてのモデルとデータセットにおいて最大の強化効果を達成している。そして、 f の値が増えれば増えるほど、エンハンスメント効果は小さくなる。

表 5 の実験結果により、増幅率 $z = 2.0$ の場合、両方のデータセットにおいて評価指標に大きな改善が見られる。Gowalla データセットでは、 $z = 1.5$ と比較し、評価指標の向上が全体的にやや大きくなっている。また、第 1 層の強化効果が最も明らかな $z = 1.5$ の実験とは異なり、 $z = 2.0$ の実験では 2 層目の強化効果が最も明らかなである。

表 6 の実験結果により、増幅率 $z = 2.5$ の場合、Movielens データセットは評価指標に大きな改善が見られる。Gowalla データセットは $K = 2$ と $K = 3$ の時にも改善が見られるが、 $K = 4$ の際に改善が見られない。そして、Movielens データセットは、増幅率 $z = 2.5$ において平均的な性能向上は、最も顕著である。

表 7 の実験結果により、増幅率 $z = 3.0$ の場合、Movielens データセットは評価指標にまだ大きな改善が見られる。Gowalla データセットでは全く改善されず、モデル性能は大幅に低下した。Gowalla データセットにとって、この増幅率は、モデルの効果を向上させる範囲を超えている。データセットによって、適切な増幅率の範囲は同じではないことがわかる。

表 5 増幅率 $z = 2.0$ の場合に各層を単層増強した結果

	Movielens		Gowalla	
	recall	ndcg	recall	ndcg
K=2				
NGCF	0.04807	0.09665	0.15515	0.21447
強化層 $f=1$	0.05149	0.10264	0.16367	0.23325
強化層 $f=2$	0.05829	0.10733	0.16074	0.23082
%Improve(max)	21.26%	11.05%	5.49%	8.76%
K=3				
強化層 $f=1$	0.05068	0.09774	0.16447	0.23404
強化層 $f=2$	0.04851	0.09612	0.16197	0.23024
強化層 $f=3$	0.04802	0.09398	0.15752	0.22867
%Improve(max)	4.00%	-2.77%	5.65%	4.62%
K=4				
強化層 $f=1$	0.05101	0.10822	0.15809	0.21920
強化層 $f=2$	0.05420	0.10412	0.16077	0.22675
強化層 $f=3$	0.05253	0.09669	0.15520	0.22076
強化層 $f=4$	0.05318	0.10372	0.15414	0.21860
%Improve(max)	7.58%	7.74%	4.05%	1.31%

表 6 増幅率 $z = 2.5$ の場合に各層を単層増強した結果

	Movielens		Gowalla	
K=2	recall	ndcg	recall	ndcg
NGCF	0.04807	0.09665	0.15515	0.21447
強化層 f=1	0.05481	0.11234	0.15417	0.20676
強化層 f=2	0.05196	0.09524	0.16398	0.23205
%Improve(max)	14.02%	16.23%	5.69%	8.20%
K=3	recall	ndcg	recall	ndcg
強化層 f=1	0.05143	0.10833	0.14921	0.19267
強化層 f=2	0.05370	0.10080	0.16326	0.22501
強化層 f=3	0.05090	0.09556	0.15392	0.21528
%Improve(max)	10.20%	7.77%	4.88%	0.58%
K=4	recall	ndcg	recall	ndcg
強化層 f=1	0.05596	0.11288	0.14399	0.18187
強化層 f=2	0.05076	0.09960	0.15144	0.20434
強化層 f=3	0.05771	0.10796	0.15045	0.20571
強化層 f=4	0.05246	0.10028	0.15003	0.20748
%Improve(max)	14.55%	12.37%	-1.99%	-7.30%

表 7 増幅率 $z = 3.0$ の場合に各層を単層増強した結果

	Movielens		Gowalla	
K=2	recall	ndcg	recall	ndcg
NGCF	0.04807	0.09665	0.15515	0.21447
強化層 f=1	0.04416	0.09774	0.14656	0.18068
強化層 f=2	0.05626	0.11170	0.13423	0.16660
%Improve(max)	17.04%	15.57%	-5.54%	-15.76%
K=3	recall	ndcg	recall	ndcg
強化層 f=1	0.04989	0.09935	0.14377	0.17664
強化層 f=2	0.05356	0.09905	0.13610	0.17066
強化層 f=3	0.05305	0.10469	0.12935	0.16332
%Improve(max)	9.91%	4.15%	-7.64%	-21.04%
K=4	recall	ndcg	recall	ndcg
強化層 f=1	0.05163	0.09908	0.13921	0.17091
強化層 f=2	0.05286	0.10316	0.13375	0.16469
強化層 f=3	0.05303	0.10631	0.12929	0.16121
強化層 f=4	0.05025	0.09527	0.13049	0.16902
%Improve(max)	5.26%	5.83%	-9.90%	-23.64%

5. おわりに

5.1 まとめ

本研究では、NGCF モデルに存在している積層問題を指摘した。そして、この積層問題を解決するために、Attention 機構の導入と単層強化という 2 つの改善手法を提案した。実験から、Attention 機構を導入する手法は Movielens データセットにおいて NGCF モデルよりも最大 5.67% の推薦精度を向上させ、Gowalla データセットにおいては有意に向上させなかった。小さいデータセットでは効果的だが、大きなデータセットではモデルの学習が複雑になるため、あまり効果がないと考えられる。単層強化の手法は実験から、Movielens データセットにおいて NGCF モデルよりも最大 21.26% の改善が示され、Gowalla データセットにおいては最大 8.76% の改善が見られた。そして、浅い畳み込み層を増強することで、モデルの効果を一気に高められる結果になった。また、増幅率 z には最適範囲があるということもわかった。詳しく述べると、Movielens データセットにおいて増幅率 $z = [1.5, 2.0, 2.5, 3.0]$ の範囲では単層強化がうまく機能した。Gowalla データセットにおいて増幅率 z が 2.5 以上になると、単層強化がうまく機能しなかった。これらによりデータセットによって有効増幅率の間隔が異なることもわかった。

以上をまとめると、本研究の主な貢献は以下の通りである。

- (1) NGCF モデルにおける積層問題を指摘し、重み分布の違いによってモデルの性能が影響を受けることを示した。
- (2) 積層問題解決のために NGCF モデルに Attention 機構

の導入を提案した。これは、Attention Network を用いてノードの層間重みの組み合わせを学習する方略をとる。実験の結果、推薦性能が向上する場合があることを示した。

(3) 積層問題解決のためのもう 1 つの手法として、単層増強を提案した。これは、1 層のみ強化するトレーニング手法で、実装は簡単である。実験の結果は、この手法は二つのデータセットでそれぞれ最大 21.26% と 8.76% の精度向上が得られた。

5.2 今後の課題

今回利用したデータセットでは結果が異なっていた。これはデータのサイズや密度の影響を受けるのか確認する必要がある。もしくは、今回導入した Attention 機構は訓練にある程度影響を与え、学習の難易度を上げると同時に、出力にも影響を与えているという手法の課題点の可能性もある。また、単層強化手法は今回の実験では効果を出したが、どこまで一般化できるかは、さらなる調査が必要である。

さらに、各ユーザーとアイテムのペアの重み分布と、層間の重み分布の組み合わせも検討する価値がある。また、増強係数 z の自動学習、異なるデータセットの統計情報に基づいて、ハイパーパラメータのチューニングの方向性を決めることも今後の発展を考えると取り上げたい課題である。

参考文献

- [1] Badrul Sarwar, George Karypis, Joseph Konstan, John Riedl. 2001. Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th international conference on World Wide Web. 285-295.
- [2] Hanwang Zhang, Fumin Shen, Wei Liu, Xiangnan He, Huanbo Luan, Tat-Seng Chua. 2016. Discrete collaborative filtering. In Proceedings of the 39th International ACM SIGIR conference on

- Research and Development in Information Retrieval. 325-334.
- [3] Xiangnan He, Hanwang Zhang, Min-Yen Kan, Tat-Seng Chua. 2016. Fast matrix factorization for online recommendation with implicit feedback. In Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval. 549-558.
- [4] Yehuda Koren. 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. 426-434.
- [5] Xiangnan He, Zhankui He, Jingkuan Song, Zhengguang Liu, Yugang Jiang, Tat-Seng Chua. 2018. Nais: Neural attentive item similarity model for recommendation. In IEEE Transactions on Knowledge and Data Engineering. 2354–2366.
- [6] Shiwen Wu, Fei Sun, Wentao Zhang, Bin Cui. 2021. Graph neural networks in recommender systems: a survey. In arXiv preprint arXiv:2011.02260.
- [7] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, Tat-Seng Chua. 2019. Neural graph collaborative filtering. In Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval. 165-174.
- [8] Jinbo Song, Chao Chang, Fei Sun, Xinbo Song, Peng Jiang. 2020. NGAT4Rec: Neighbor-Aware Graph Attention Network for Recommendation. In arXiv preprint arXiv:2010.12256.
- [9] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval. 639-648.
- [10] Xiang Wang, Hongye Jin, An Zhang, Xiangnan He, Tong Xu, Tat-Seng Chua. 2020. Disentangled Graph Collaborative Filtering. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. 1001-1010.
- [11] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, Tat-Seng Chua. 2019. Kgat: Knowledge graph attention network for recommendation. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 950-958.
- [12] Wentao Zhang, Ziqi Yin, Zeang Sheng, Wen Ouyang, Xiaosen Li, Yangyu Tao, Zhi Yang, Bin Cui. 2021. Graph attention multi-layer perceptron. In arXiv preprint arXiv:2108.10097.
- [13] Qimai Li, Zhichao Han, Xiao-Ming Wu. 2018. Deeper insights into graph convolutional networks for semi-supervised learning. In Thirty-Second AAAI conference on artificial intelligence.
- [14] Yixin Cao, Xiang Wang, Xiangnan He, Zikun Hu, Tat-Seng Chua. 2019. Unifying Knowledge Graph Learning and Recommendation: Towards a Better Understanding of User Preferences. In The world wide web conference. 151-161.
- [15] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural Collaborative Filtering. In Proceedings of the 26th international conference on world wide web. 173-182.
- [16] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, Lars Schmidt-Thieme. 2012. BPR: Bayesian Personalized Ranking from Implicit Feedback. In arXiv preprint arXiv:1205.2618.
- [17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin. 2017. Attention is all you need. In Advances in neural information processing systems. 5998-6008.
- [18] Jingyuan Chen, Hanwang Zhang, Xiangnan He, Liqiang Nie, Wei Liu, Tat-Seng Chua. 2017. Attentive Collaborative Filtering: Multimedia Recommendation with Item-and Component-Level Attention. In Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval. 335-344.