The 33rd Annual Conference of the Japanese Society for Artificial Intelligence, 2019

# グラフ畳み込み層を有する敵対的生成ネットワークによる推薦システムの提案

Recommendation System based on Generative Adversarial Network with Graph Convolutional Layers

笹川 天斗 *1 Takato Sasagawa　河合 新 *2 Shin Kawai　延原 肇 *2 Hajime Nobuhara

*1筑波大学理工学群工学システム学類
College of Engineering Systems, School of Science and Engineering, Univ. of Tsukuba

*2筑波大学大学院システム情報工学研究科知能機能システム専攻
Dept. of Intelligent Interaction Technologies, Graduate School of Systems and Information Engineering, Univ. of Tsukuba

A Graph Convolutional Generative Adversarial Network (GCGAN) is proposed to effectively recommend to new users or items. To maintain scalability, the discriminator is improved to capture latent features of users and items by using graph convolution from a minibatch size bipartite graph. Through the experiment using MovieLens dataset, it is confirmed the effectiveness of the proposed GCGAN compared with the conventional methods.

## 1. はじめに

近年の深層学習において、画像生成分野で注目されてきた生成モデルである GAN(Generative Adversarial Networks)[Goodfellow 14] が推薦システムにおいても有効であることが示されている [Wang 17b, Wang 17a, Chae 18]。この GAN による推薦システムは、推薦する上でユーザの既存の購入や評価の履歴を必要としない。よって、協調フィルタリング等の履歴の類似度で推薦をするアルゴリズムが抱えるユーザーの評価が極端に少ない、あるいは無いといった状況下で推薦ができないコールドスタート問題を解消することができる。一方で、履歴の代わりに使用するユーザーおよびアイテムのドメイン情報 (ユーザーの年齢やアイテムのジャンルなどの属性情報) の両方を同時かつ効果的に利用することのできるモデルは未だ存在しない。

これに対して、ソーシャルネットワーク、知識グラフ、タンパク質相互作用ネットワークなどのグラフデータのための強力なツールとして、グラフ畳み込みネットワーク [Kipf 16] が提案されている。推薦においても、ユーザーとアイテムの相互作用の構造を 2 部グラフとみなすことで応用できる。情報推薦に 2 部グラフを用いる研究では、グラフ畳み込みを用いて与えられたグラフ構造を再構成するモデルの有効性が示されている [Berg 17]。グラフ畳み込みでは、畳み込みをおこなうことでノード間の特徴量が集約していき、大局的な特徴量を学習できる。しかし、ノードが持つ特徴量をエッジにしたがって伝搬させるため、ノード数が固定されており、新規ノードを学習したモデルに追加するには、モデルを再学習する必要がある。

本研究では、ユーザとアイテムの潜在的特徴をグラフ構造を介して学習するグラフ畳み込み敵対的生成ネットワーク (GCGAN:Graph Convolutional Generative Adversarial Network, 図 1) を提案する。GCGAN は生成器の入力ノイズにラベル情報を追加することで生成データを制御する CGAN[Mirza 14] に基づいている。ラベル情報としてユーザーの特徴量ベクトル (ドメイン情報) を入力し、ユーザーのアイテム評価・嗜好を表す 2 部グラフを生成する。このグラフで、従来のモデルではできなかったユーザーとアイテム両方のドメイン情報を活用し、推薦性能を向上させる。またノード数において、バッチサイズごとにグラフを統合することにより、新規ノードに対しても大局的な特徴量を学習できるメリットを確保し、モデルを再構築せずとも新規ノードに対して予測が可能となる。これにより、ユーザーとアイテム両方のドメイン情報を活用し、コールドスタート問題を解消できる。

連絡先: 笹川 天斗，筑波大学理工学群工学システム学類，sasagawa@cmu.iit.tsukuba.ac.jp

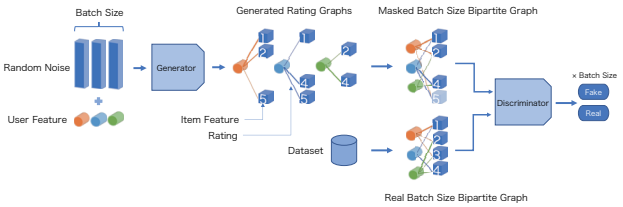


図 1: GCGAN フレームワークの全体図

提案する GCGAN の有効性を示すために、映画評価履歴のデータセット（MovieLens100K, MovieLens1M）[Harper 16] を用いた実験を行う。比較対象として GAN ベース推薦システムである CFGAN[Chae 18] と Precision@N(Top-N の Precision) の結果を比較し、GCGAN の推薦システムにおける有効性を示す。

## 2. 関連研究

### 2.1 敵対的生成ネットワーク (GAN) を用いた推薦

GAN[Goodfellow 14] を利用し情報検索と推薦に応用した最初のモデルとして IRGAN[Wang 17b] がある。この研究では、Web 検索、アイテムの推薦、質問の回答のなどの情報検索タスクでその機能を実証している。同じく GraphGAN[Wang 17a] と CFGAN[Chae 18] は、GAN が推薦システムで成功する可能性を実証した先駆的な方法である。これらは生成器 $G$ にユーザが購入するアイテムの予測リスト (ベクトル) を生成させ、識別器 $D$ が学習データ上の真のリストか、生成された偽物かを分類することで、生成器 $G$ がユーザー嗜好の真の分布をとらえる手法である。GAN に基づく生成モデルによる推薦は、協調フィルタリング等の評価履歴の類似度から予測するのではなく、嗜好の分布を学習するためコンテンツベースアルゴリズ

ムの一種であるといえる。

### 2.2 2部グラフの推薦システム

グラフを対象とした深層学習の研究に基づき、行列補間のための自動エンコーダフレームワークを推薦に応用する研究が行われている [Berg 17]。具体的には、ユーザーとアイテムをノードとし、嗜好・評価のリンクで表した2部グラフ構造に置き換え、推薦をグラフのリンク予測問題とみなしている。そしてグラフ畳み込みネットワークにより、ユーザーとアイテムの潜在表現を学習し、再構築したグラフを用いて推薦を行う。推薦するグラフがソーシャルネットワークのようなドメイン情報を伴う場合に特に有効であり、情報推薦においては、グラフとして表現される評価値データとドメイン情報などの外部情報を組み合わせることで、評価値が少ない状態 (コールドスタート) での性能向上が見込まれている。

## 3. 提案手法

提案法は、ユーザーとアイテムの評価（嗜好）関係を表す2部グラフを生成するネットワーク (生成器 $G$) を学習することで実現される。生成器に入力するのはユーザーの特徴量と正規分布に従うノイズで、出力は2部グラフである。この2部グラフをバッチサイズ個分を一つの 2 部グラフに統合し識別器に入力する。識別器では入力された 2 部グラフがデータセット上の真のグラフか、生成された偽物かを分類するように学習する。提案法の識別器では、評価値に加えてドメイン情報も扱うグラフ畳み込みを用いることで、大局的な特徴量が学習され、分類性能の向上が見込まれる。したがって、敵対的に学習される生成器についても生成性能 (推薦性能) の向上が期待される。

### 3.1 GCGAN フレームワーク

GCGAN における生成データはユーザーノード $u_i \in \mathcal{U}, i \in \{1, ..., N_u\}$ に対応するアイテムノード $v_j \in \mathcal{V}, j \in \{1, ..., N_v\}$ とのエッジの接続関係 $m_i \in \mathbb{R}^{N_v}, i \in \{1, ..., N_u\}$ である。エッジは予測されるユーザーとアイテムの評価関係を表し、嗜好の度合いを示す重み値が付随する。また各ノードには $F_u$, $F_v$ を次元数とする特徴量 $x_u^{(i)} \in \mathbb{R}^{F_u}$ , $i \in \{1, ..., N_u\}$, $x_v^{(j)} \in \mathbb{R}^{F_v}$ , $j \in \{1, ..., N_v\}$ が付随している。したがって GCGAN はユーザーとアイテムによる二部グラフのサンプルを生成器 $G$ で生成し、生成したグラフをもっともらしい評価関係になるよう識別器 $D$ がデータセットを用いて識別しながら学習する。GCGAN は

$$\min_G \max_D V(D, G) =$$
$$\mathbb{E}_{m \sim p_{data}(m)}[\log D(m \otimes (x_u, x_v))] +$$
$$\mathbb{E}_{z \sim p_z(z)}[\log(1 - D((G(z|x_u) \odot e) \otimes (x_u, x_v)))],$$

をもとに関数 $D$ と $G$ を引数に取る価値関数 $V(D, G)$ によって定義され、識別器 $D$ について最大、生成器 $G$ について最小となるような $D$, $G$ を定める。ここでの $(\otimes)$ はグラフ畳み込みを、$(\odot)$ はベクトルの要素ごと積を意味している。

生成器 $G$ は $\theta_G$ をパラメータとして、標準正規分布 $p_z$ に従うノイズ $z \in \mathbb{R}^n$ を入力し、出力するエッジの接続関係 $m$ の生成器上での分布 $p_g$ を学習するため、ノイズ分布 $p_z(z)$ をデータ空間へ写像する関数

$$m = G(z|x_u; \theta_G) \ (\in \mathbb{R}^{N_v}),$$

となる。このときユーザーの特徴量 $x_u$ をノイズ $z$ に結合して入力することで条件付きモデルに拡張する。したがって生成器はユーザー特徴量とノイズの分布から生成されるデータの分布を学習することが期待される。

そして識別器 $D$ は $\theta_D$ をパラメータとして、入力 $m$ が $p_g$ からではなく、訓練データ上のものである確率 $y$ を出力する関数

$$y = D(m \otimes (x_u, x_v); \theta_D) \ (\in \mathbb{R}),$$

とする。生成器から生成されたグラフは識別器 $D$ へ入力される前に $e \in \mathbb{R}^{N_v}$ によってマスクされる。マスク $e$ は生成器に入力されたユーザー特徴に対応する訓練データの欠損値を、生成されたグラフにおいて再現する。識別器 $D$ ではユーザーとアイテムの特徴量 $x_u, x_v$ 両方を使用してグラフ畳み込みを行い、畳み込み後の特徴量と接続関係 $m$ を結合して入力することで、条件付きモデルとして拡張する。

価値観数 $V$ は Algorithm 1 によって最適化される。まずバッチサイズ個 $b$ のノイズ $z$ を $p_z(z)$ から取り出す。同様にグラフの接続関係 $m$ を $p_{data}(m)$ から取り出す。それらをユーザー特徴量 $x_u$ と結合する。生成器によって生成されたグラフに対応する真のグラフの欠損値をマスクする。その後 $\theta_D$ における確率的勾配が上がるように識別器 $D$ を更新する。次にバッチサイズ個のノイズ $z$ を取り出し、$\theta_G$ における確率的勾配が下がるように生成器 $G$ を更新する。これを訓練回数分だけ繰り返すことで最適化される。

### 3.2 識別器におけるバッチサイズ統合とグラフ畳み込み層

GCGAN において識別器 $D$ が識別するのは、ユーザーとアイテムの評価・嗜好関係を表す 2 部グラフである。したがって識別器 $D$ は入力された 2 部グラフが、データ上のものか生成されたものかを識別する。エッジの接続関係による類似ノード間の特徴量の学習のためには、複数のユーザーとアイテムのノードが必要だが、生成器 $G$ は一人のユーザー $u_i$ のみに関するアイテムの接続関係 $m_i$ を出力する (図 1 の Generated Rating Graphs)。したがって、提案法ではバッチサイズ $b$ 人分の 2 部グラフを統合し、識別器 D への入力とした。この時、識別器 $D$ がユーザーノードが持つ特徴量 $x_u^{(i)} \in \mathbb{R}^{F_u}, i \in \{1, ..., b\}$ とアイテムノードのもつ特徴量 $x_v^{(j)} \in \mathbb{R}^{F_v}, j \in \{1, ..., N_v\}$ を学習するためグラフ畳み込み層を使用する。この畳み込み層は任意の数で多層化でき、特徴量が各ノードを伝搬し大域的な特徴量を学習する。識別器 D はこの畳み込んだ $h_u^{(i)}$ と隣接関係を表す $m^{(i)}$ を結合し、全結合層にてグラフが本物か偽物かを識別する。

表 1: データセット

| Dataset | Users | Items | Ratings | UF | IF |
|---|---|---|---|---|---|
| ML100K | 948 | 1,683 | 100,000 | 23 | 18 |
| ML1M | 6,040 | 3,706 | 1,000,000 | 23 | 18 |

## 4. 評価実験

### 4.1 実験環境

データセットはユーザーが映画に対して評価をしている 2 つのサイズの MovieLens[Harper 16] データセット (表 1) を使用する。このデータセットには各ユーザーが各アイテムに行った評価の値以外に、評価を行ったユーザーとアイテムのドメイン情報が付随している。実験ではドメイン情報としてユーザーの

**Algorithm 1** Minibatch stochastic gradient descent training of graph convolutional generative adversarial nets.

**for** number of training iterations **do**

- Sample minibatch of $b$ noise sample $\{z^{(1)}, \ldots, z^{(b)}\}$ from $p_z(z)$.
- Sample minibatch of $b$ examples $\{m^{(1)}, \ldots, m^{(b)}\}$ from data generating distribution $p_{data}(m)$.
- Sample minibatch of $b$ user feature $\{x_u^{(1)}, \ldots, x_u^{(b)}\}$.
- Sample minibathc of $b$ mask $\{e^{(1)}, \ldots, e^{(b)}\}$.
- Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_D} = \frac{1}{b}\sum_{i=1}^{b}[\log D(m^{(i)} \otimes (x_u^{(i)}, x_v)) + \log(1 - D((G(z^{(i)}|x_u^{(i)}) \odot e^{(i)}) \otimes (x_u^{(i)}, x_v)))].$$

- Sample minibatch of $b$ noise sample $\{z^{(1)}, \ldots, z^{(b)}\}$ from $p_z(z)$.
- Sample minibatch of $b$ user feature $\{x_u^{(1)}, \ldots, x_u^{(b)}\}$.
- Update the generator by ascending its stochastic gradient:

$$\nabla_{\theta_G} = \frac{1}{b}\sum_{i=1}^{b}[\log(1 - D((G(z^{(i)}|x_u^{(i)}) \odot e^{(i)}) \otimes (x_u^{(i)}, x_v)))].$$

**end for**

The gradient-based updates can use any standard gradient-based learning rule.

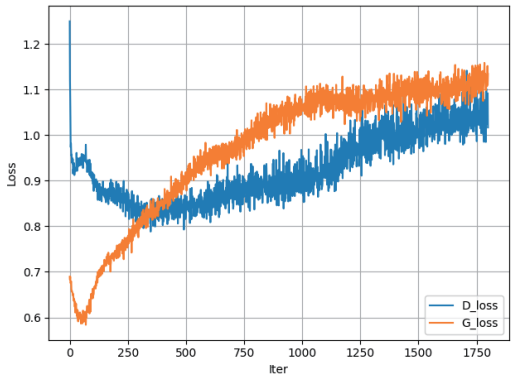


図 2: GCGAN の学習中のミニマックスゲームの推移の様子

年齢・性別・職業のデータと、アイテムのジャンルのデータを使用した。表中の UF(User Feature) と IF(Item Feature) は使用した特徴量の次元数である。MovieLens100K,1M の評価値は $r = \{\text{Null}, 1, 2, 3, 4, 5\}$ を持ち、値が大きいほど高評価をしている。データセットを (ユーザー数 $N_u$ ×アイテム数 $N_v$) で要素に評価値を持つ評価行列に変換し、評価なしの欠損値 (Null) は 0 とする。評価行列をユーザーの軸で 8:2 に分割し、訓練データとテストデータとした。訓練データでの学習後、生成器にテストデータ内のユーザー特徴量を入力し、生成した二部グラフのエッジ上の重み上位 N 個を推薦アイテムとする。この推薦アイテムと評価値 4 以上を嗜好ありとしたテストデータにより推薦の性能を Precision@N にて評価する。同じ GAN を用いた推薦システムであり、ユーザー特徴量を入力とした生成器と識別器を多層ニューラルネットによって構成している CFGAN を比較対象に設定する。

### 4.2 結果・考察

図 2 に学習における生成器と識別器の損失の推移を示す。通常の GAN[Goodfellow 14] と同様に GCGAN も生成器と識別器がシンクロして学習を進めていることが確認できる。

次に MovieLens100K と MovieLens1M での実験結果を表 2 に示す。GCGAN は MovieLens100K においてバッチサイズ 64,128 両方で比較手法を下回り、MovieLens1M においては両方のバッチサイズで比較手法を上回る結果となった。MovieLens100K で比較手法に劣り、MovieLens1M で比較手法より向上した理由として、データセットのサイズが MovieLens100K では小さく、グラフ畳み込みを行うことで識別するノードの特徴量が均一化・平滑化してしまったのではないかと考えられる。対して MovieLens1M では、十分なノード数を確保できているため、グラフ畳み込みが効果的にノードの特徴量を学習できたと考えられる。

GCGAN ではバッチサイズ個のノードを統合し、その 2 部グラフ上で畳み込みを行い特徴量を学習する。したがってバッチのサイズによってグラフ上での特徴量を学習する領域が決まり、その領域を大きくするほど大局的な特徴量を学習し、識別性能を介した推薦性能の向上が期待される。図 3 に MovieLens100K,1M においてバッチサイズを段階的に変更した結果を示す。バッチサイズを大きくすることにより性能がよくなっていることが確認できる。またある程度を超えると、そこから性能が悪化することが確認できた。これは畳み込むグラフサイズが、最終的に識別するユーザー一人に対するグラフに対して大きくなりすぎると、特徴量が平滑化し局所性が無視され識別が困難になるためであると考えられる。実際に図 3 より、アイテム数が少ない MovieLens100K では MovieLens1M よりも早い段階で性能が悪化していることが確認できる。

グラフ畳み込み層は任意の数に多層化でき、一層の畳み込みでアイテムノードの特徴量をユーザーノードへ、ユーザーノードの特徴量をアイテムノードへ伝搬する。したがって直接接続していないノードにも、多層化によりエッジを介し特徴量が伝搬される。MovieLens100K,1M においてグラフ畳み込み層の層数と評価値の推移を図 4 に示す。図より MovieLens1M においては 5 層目まで畳み込み層の層数が多くなるほど向上していることが確認できる。MovieLens100K では畳み込み層の層数による推薦性能の向上は確認できなかった。グラフが小さく、一回の畳み込みで平滑化が進んでしまったためであると考えられる。

表 2: MovieLens100K,1M での Precision@N の実験結果

| Dataset | MovieLens 100K | | | | | | MovieLens 1M | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bache Size | 64 | | | 128 | | | 64 | | | 128 | | |
| method | P@5 | P@10 | P@15 | P@5 | P@10 | P@15 | P@5 | P@10 | P@15 | P@5 | P@10 | P@15 |
| CFGAN | **0.610** | **0.678** | **0.684** | **0.667** | **0.768** | **0.808** | 0.578 | 0.735 | 0.759 | 0.662 | 0.732 | 0.757 |
| GCGAN | 0.499 | 0.611 | 0.648 | 0.425 | 0.561 | 0.618 | **0.677** | **0.749** | **0.771** | **0.705** | **0.765** | **0.784** |

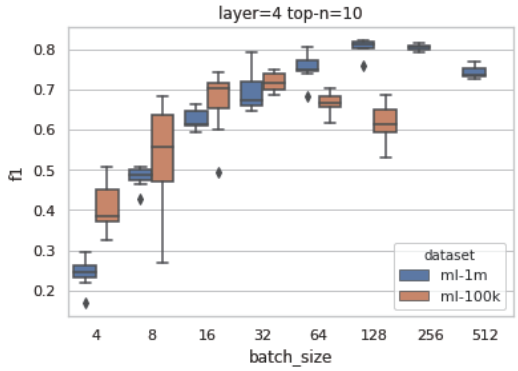


図 3: バッチサイズと f1 スコアの推移

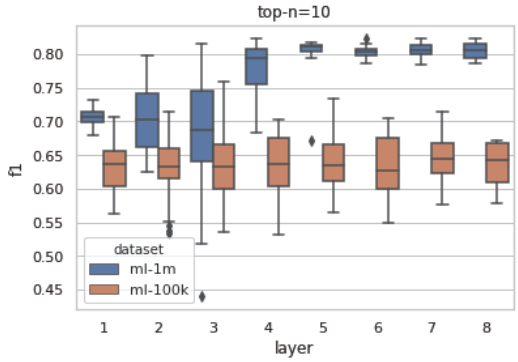


図 4: グラフ畳み込み層の層数と f1 スコアの推移

## 5. おわりに

本研究では推薦システムにおけるコールドスタート問題を解消するため GCGAN を提案した。GAN とグラフ畳み込みを組み合わせた GCGAN は、グラフ構造を介した特徴量の伝搬によりドメイン情報を効果的に学習でき、ユーザー特徴のみで推薦するため新規ノード (ユーザー) に対して再学習する必要がない。GCGAN は MovieLens1M のデータセットにおいて比較手法よりも推薦性能が向上した。またバッチサイズ個ごとのノードの統合と、グラフ畳み込み層の層数における推薦性能への効果を確認した。今後の課題として、MovieLens よりユーザーとアイテムの特徴より多いデータセットに GCGAN を適用すること、小さなデータセットへの対策、生成器においてもドメイン情報を考慮できるモデルを構築していく必要があると考えている。

## 参考文献

[Berg 17] Berg, van den R., et al.: Graph Convolutional Matrix Completion, *stat*, Vol. 1050, p. 7 (2017)

[Chae 18] Chae, D.-K., et al.: CFGAN: A Generic Collaborative Filtering Framework based on Generative Adversarial Networks, in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pp. 137–146ACM (2018)

[Goodfellow 14] Goodfellow, I., et al.: Generative adversarial nets, in *Advances in neural information processing systems*, pp. 2672–2680 (2014)

[Harper 16] Harper, F. M., et al.: The movielens datasets: History and context, *Acm transactions on interactive intelligent systems (tiis)*, Vol. 5, No. 4, p. 19 (2016)

[Kipf 16] Kipf, T. N., et al.: Semi-supervised classification with graph convolutional networks, *arXiv preprint arXiv:1609.02907* (2016)

[Mirza 14] Mirza, M., et al.: Conditional generative adversarial nets, *arXiv preprint arXiv:1411.1784* (2014)

[Wang 17a] Wang, H., et al.: Graphgan: Graph representation learning with generative adversarial nets, *arXiv preprint arXiv:1711.08267* (2017)

[Wang 17b] Wang, J., et al.: Irgan: A minimax game for unifying generative and discriminative information retrieval models, in *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*, pp. 515–524ACM (2017)