

データセットを用いずに任意の特徴量を満たすグラフを生成する Deep Learning 手法に関する研究

井脇 蒼葉[†] 津川 翔^{††} 眞田 亜紀子[†] 渡部 康平^{†††}

[†] 長岡技術科学大学 大学院工学研究科
〒940-2188 新潟県長岡市上富岡町 1603-1

^{††} 筑波大学 システム情報系
〒305-8573 茨城県つくば市天王台 1-1-1

^{†††} 埼玉大学 大学院理工学研究科
〒338-8570 埼玉県さいたま市桜区 下大久保 255

E-mail: [†]s223094@stn.nagaokaut.ac.jp, [†]amanada@vos.nagaokaut.ac.jp,
^{††}s-tugawa@cs.tsukuba.ac.jp, ^{†††}kwatabe@mail.saitama-u.ac.jp

あらまし 近年、ソーシャルネットワークの解析や分子探索のために、グラフの数学的な量 (特徴量) を制御して所望のグラフを生成する Deep Learning 手法が提案されている。しかし、これらの Deep Learning 手法には、大規模なデータセットを必要とすることや、生成されるグラフの特徴量の範囲に数学的な保証がないといった課題が存在する。本稿では、古典的なグラフ生成手法である Barabási-Albert (BA) Model を基盤とし、データセットを必要とせずに任意の値の特徴量を指定してグラフを生成するための Deep Learning 手法を提案する。提案手法により、極めて高い精度でクラスタ係数を指定してグラフを生成することが可能であり、生成されたグラフは BA Model の特性であるスケールフリー性を概ね維持することを確認した。

キーワード ネットワーク, グラフ, LSTM, 条件付き生成, 機械学習

A Study on Machine Learning Approach for Creating Graphs with Given Graph Statistics

Aoba IWAKI[†], Sho TSUGAWA^{††}, Akiko MANADA[†], and Kohei WATABE^{†††}

[†] Graduate School of Engineering, Nagaoka University of Technology
1603-1 Kamitomioka, Nagaoka, Niigata, 940-2188 Japan

^{††} Institute of Systems and Information Engineering, University of Tsukuba 1-1-1, Tennohdai, Tsukuba, Ibaraki
305-8573, Japan

^{†††} Graduate School of Science and Engineering, Saitama University
255, Shimo-Okubo, Sakura-ku, Saitama, Saitama 338-8570, Japan

E-mail: [†]s223094@stn.nagaokaut.ac.jp, [†]amanada@vos.nagaokaut.ac.jp,
^{††}s-tugawa@cs.tsukuba.ac.jp, ^{†††}kwatabe@mail.saitama-u.ac.jp

Abstract In recent years, machine learning techniques have been proposed to analyze the mathematical properties of graphs and create graphs for applications such as social network analysis and molecular discovery. However, these machine learning methods face challenges, including the need for large datasets and the lack of mathematical guarantees on the range of features of the generated graphs. This paper proposes a machine learning-based graph analysis founded on the classical Barabási-Albert (BA) model. We confirmed that the proposed method can generate graphs by specifying cluster coefficients with extremely high accuracy, and that the generated graphs generally maintain the scale-free property of the BA Model.

Key words Network, Graph, LSTM, Conditional Generation, Machine Learning

1. はじめに

グラフは通信ネットワークやソーシャルネットワーク、化学分野で広く用いられており、ネットワークや分子の構造を分析するための有用なツールである。例えば、ソーシャルネットワークの構造からコミュニティを検出したり、化学物質の構造から物性を予測したりするといった応用が存在する。そして、特定の構造が全体の性質にどのように影響するかを研究し、所望のネットワークや分子構造を生成することは重要な課題である。これが可能になればプライバシー上の理由から取得できない人間関係のネットワークを擬似的に生成し解析に用いたり、所望の性質を持つような化合物を生成することが可能になる。

近年、グラフの持つ数学的な量(以後、グラフの特徴量と呼ぶ)を制御して所望のグラフを生成するグラフ生成技術が注目を集めている。グラフ生成研究には、主に2つのアプローチがあり、1つは確率的にノードとエッジを選択するアルゴリズムによってグラフを生成する手法(以後、ランダムベースの手法と呼ぶ)であり、もう1つはDeep Learning手法を用いたグラフ生成手法(以後、Deep Learning手法と呼ぶ)を用いたグラフ生成手法である。ランダムベースの手法は、生成過程を数学的に解析することで生成されるグラフの一部の特徴量の範囲や平均値などを保証することができるが、無数にあるグラフの特徴量のほとんどを制御することはできない。Deep Learning手法は、多様な特徴量を制御するポテンシャルを持っている一方で、学習や生成の過程はブラックボックスであるため生成されるグラフの特徴量について数学的な保証は一般に存在しない。そのため、ランダムベースの手法の厳密性とDeep Learning手法の柔軟性を併せ持つ手法が求められている。

ランダムベースの手法の代表的なモデルとして、Erdős-Rényi (ER) Model [1] や Barabási-Albert (BA) Model [2] が挙げられる。これらのモデルは、それぞれ特定の特徴量を制御することを得意としている。ER Model はグラフの平均次数を制御することができ、BA Model は生成したグラフの次数列のべき指数の平均値が3となることが知られている。ここで次数分布とは、横軸に次数 k 、縦軸にノード数をとったもので、べき指数が3の場合、次数 k のノードの数が k^{-3} に比例することを意味する。一般に、ランダムベースの手法はDeep Learning手法と比較して単純な処理であることから高速にグラフを生成することができ、データセットも必要としない。一方で、同じモデルを用いて元々制御対象とはしていなかったグラフの特徴量を制御することはできず、それぞれ制御したいグラフの特徴量に応じてモデルを構築する必要がある。

Deep Learning手法の代表的なモデルとして、GraphVAE [3] や GraphRNN [4] が挙げられる。これらのモデルはデータセットからグラフ構造とグラフの特徴量の関係性を学習することで、様々な特徴量を持つグラフを生成する。一般に、Deep Learning手法はランダムベースの手法と比較して柔軟性が高く、様々な用途に適用することができる [5][6]。一方で、生成にかかる時間は一般にランダムベースの手法に比べて長く、大規模なデータセットを必要とすることが多い。

Deep Learning手法には、一般に以下の2つの課題がある。

- (1) 学習に大規模なデータセットを必要とすること
- (2) グラフの特徴量の範囲に数学的な保証がないこと

Deep Learning手法では多くの学習用データセットが必要であるが、それらの入手は困難であることも多い。例えば、プライバシーに関わるようなソーシャルネットワークに関するデータなどは容易に入手することはできない。また、Deep Learning手法の学習や生成の過程はブラックボックスであるため、生成されるグラフの特徴量の範囲や平均値について数学的な保証がなく、信頼性に欠けるという課題がある。

本稿では、古典的なグラフ生成手法の1つであるBA Modelを拡張し、データセットを必要とせずに任意の特徴量を持つグラフを生成するDeep Learning手法を提案する。その目的は大きく2つに分けられる。1つ目は、BA Modelを基盤とすることで、BA Modelにより生成されるグラフの次数分布のべき指数の平均値が3であるという特性を維持しつつ、他の特徴量をコントロールすること。2つ目は、Generative Adversarial Nets (GAN) [7] に似た技術を用い、2つのモデルを交互に学習することにより、データセットを必要とせずに任意の特徴量を持つグラフを生成することである。GANとは、生成器と識別器と呼ばれる2つのモデルを交互に学習させることで、生成器は本物らしいデータを生成することで識別器を騙すように学習し、識別器は与えられたデータが本物かどうか見破り、生成器に騙されないように学習することで、生成器が生成するデータが本物らしいものになるように学習する手法である。

本稿の構成は以下のとおりである。第2章では、本研究に関連する研究を紹介する。その後、第3章では提案手法の解説を行う。第4章では提案手法を用いて実験を行い、その結果を示す。最後に第5章で本研究から得られた知見について述べる。

2. 関連研究

2.1 Barabási-Albert (BA) Model

古典的なグラフ生成手法を代表する Barabási-Albert (BA)

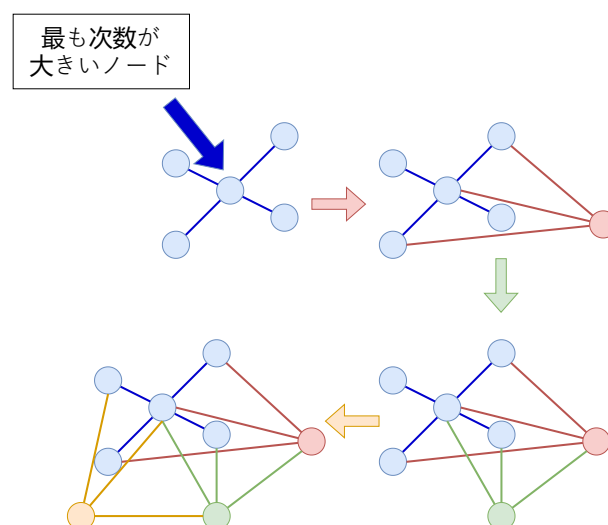


図1 Barabási-Albert (BA) model によるグラフの成長過程の例

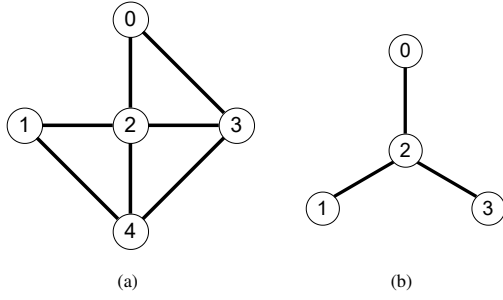


図2 グラフの例

Model[2] は、スケールフリー性を持つグラフを生成することで知られている。スケールフリー性を持つグラフとは、巨大な次数を持つノードが少数存在し、その他の多くのノードは比較的小さな次数しか持たないという性質を持ったグラフであり、数学的には次数分布がべき乗則に従うグラフのことを指す。スケールフリー性はソーシャルネットワークにおけるインフルエンサー(次数大)とそのフォロワー(次数小)など、現実世界で多くみられるネットワークの特性であるため、BA Model は実世界のネットワークをモデル化する際に広く用いられる。

BA Model は、初期グラフに新規ノードを追加し、既存ノードからランダムに m 個のノードを選択し新規ノードに接続するという操作を繰り返し、グラフを生成する(図1)。図1の例で、 m は3である。この時、既存ノードはその次数に比例した確率分布 p^{BA} から選択される。これにより、高い次数を持つノードはよりエッジを獲得しやすくなり、低い次数を持つノードは相対的にエッジを獲得しづらくなるため、スケールフリー性を持つグラフが生成される。

2.2 Deep Graph Generative Models

Deep Graph Generative Models とは、Deep Learning 手法を用いてグラフを生成する手法全般を表す名称である[8]。主な生成手法は2つある。1つは隣接行列を Multi-Layer Perceptron (MLP) に入力して隣接行列を出力する画像生成モデルに類似した方法である。もう1つはグラフを何らかのシーケンスに変換し、時系列データとして扱うことで Long Short Term Memory (LSTM)[9] などを用いてシーケンスの学習・生成を行い、生成されたシーケンスからグラフを再構築する方法である。本稿では主に後者の手法に焦点を当てる。

Deep Learning 手法を用いたグラフ生成モデルには、GraphVAE[3] や GraphRNN[4] がある。GraphVAE は、隣接行列やノードやエッジの特徴量を入力として用いる Deep Learning 手法であり、隣接行列やエッジやノードの特徴量を Variational AutoEncoder (VAE)[10] を用いて潜在空間に埋め込み、潜在空間からサンプリングした潜在変数を元にグラフを生成する。GraphRNN は、シーケンス化されたグラフを入力として用いる Deep Learning 手法であり、シーケンスを再帰的な機械学習モデルに入力し、グラフの構造を学習することで、学習データセットの特徴を再現したグラフを生成する。

3. 提案手法

3.1 BA Model を基盤としたグラフ生成手法について

提案手法では、BA Model の確率分布 p^{BA} を Deep Learning 手法を用いて補正することで目的の特徴量を持つグラフを生成する。この補正に用いる確率分布を本稿では、Enhancer と呼ぶこととする。Enhancer 付き確率分布 $p^{Modified}$ は、BA Model 本来の確率分布 p^{BA} と Enhancer の確率分布 $p^{Enhancer}$ を用いて以下の式で計算される。

$$p^{Modified} = \frac{p^{BA} + p^{Enhancer}}{2}. \quad (1)$$

ただし、 $p^{Modified}$, p^{BA} , $p^{Enhancer}$ はそれぞれ N 次の行ベクトルであり、 N は生成するグラフのノード数である。グラフの成長過程では、グラフのサイズが N に満たないためノードが存在しない部分は0で埋められるものとする。提案手法において、新規ノードと接続する既存ノードは、 $p^{Modified}$ に基づいてサンプリングされる。

3.2 交互学習について

提案手法は次ページ図3に示すように、Graph Generator 部と Feature Estimator 部と呼ばれる2つのモデルを用いて交互に学習を行う。これを本稿では、交互学習と呼ぶこととする。Graph Generator 部は、Feature Estimator 部の出力が目標とする特徴量 $F^{Specified}$ に近づくように学習を行い、Feature Estimator 部は、Graph Generator 部の出力したグラフの特徴量を正しく推定できるように学習する。Graph Generator 部の学習の際は、Graph Generator 部から目標とする特徴量の値が流入しないようにするため、Feature Estimator 部のパラメータは更新されない。Graph Generator 部と Feature Estimator 部がそれぞれ互いの出力を学習するという交互学習の手法を用いることで、データセットを必要としないのが提案手法の特徴である。

交互学習の学習単位について説明する。交互学習では、Graph Generator 部と Feature Estimator 部の学習を合わせて1つのセットとし、この学習単位を指定の回数繰り返す。Graph Generator 部の学習においては、Feature Estimator 部の出力を元に生成するグラフの特徴量が $F^{Specified}$ に近づくように指定の回数学習を行う。Feature Estimator 部の学習においては、Graph Generator 部から生成されたグラフを入力、その特徴量を正解ラベルとして指定の回数学習を行う。それぞれの学習回数はハイパーパラメータとして設定され、Graph Generator 部側は Feature Estimator 部に過度に適應しない回数にする必要があり、Feature Estimator 部は精度が収束するまで学習を続ける必要がある。

3.3 グラフのシーケンスへの変換について

機械学習モデルに入力できるのは一般にベクトルや行列である一方で、グラフはノードとエッジでの集合として表されるため、機械学習モデルに入力できる形に変換する必要がある[8]。一般には、隣接行列などに変換され計算できる形にしてから入力することが多いが、その他にもグラフをシーケンスに変換し、グラフを時系列データとして扱うことで、再帰的な機械学習モデルを用いてグラフ構造を学習する手法も多く提案されている[4]。本研究では、後者のグラフをシーケンスに変換する手

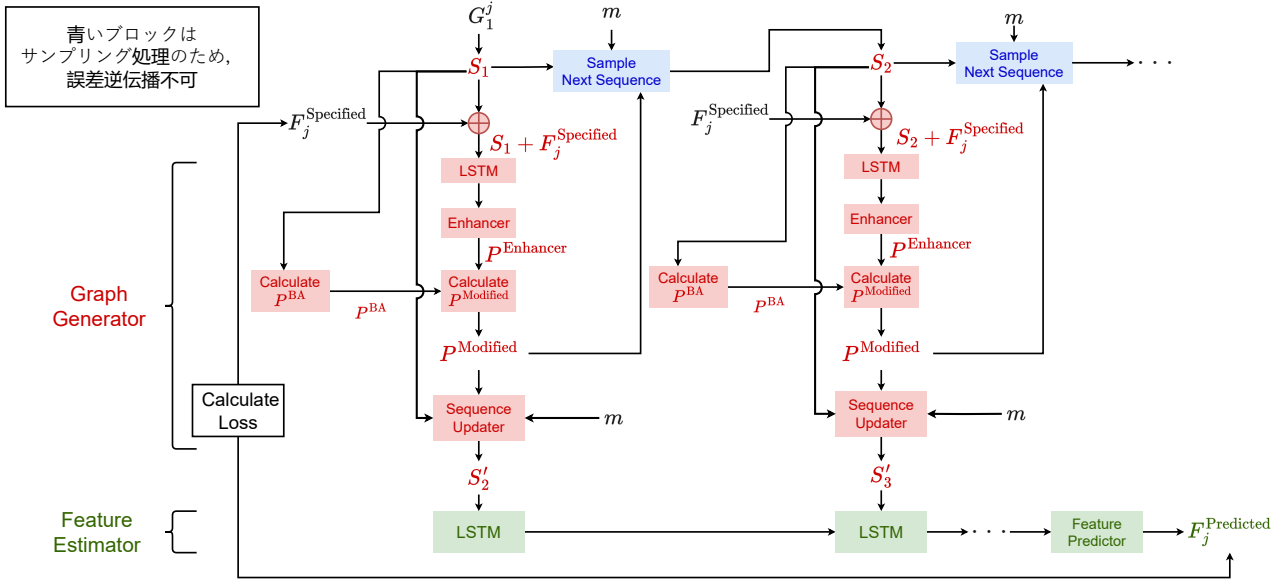


図3 提案手法の構成

法を用いてグラフの構造を学習するモデルを

本研究では、グラフをシーケンスに変換する手法として次数列の時間発展を用いることを提案する。次数列とは、ノードの次数を要素とする数列であり、図2(a)のようなグラフの場合、次数列は $[2, 2, 4, 3, 3]$ となる。ただし、図2(a)のノードの番号はノードIDを表しており、次数列は次数をノードIDが昇順となるように並べたものである。従って、一般に次数列の要素は整数である。提案手法において、グラフは初期グラフから始まり、新規ノードが追加されることで次数列が時間発展する。この次数列の時間発展にグラフの構造情報が含まれるという仮定を置き、次数列の時間発展を用いた学習モデルを提案する。

図3におけるサンプリング処理(青いブロック)は微分不可能であり、誤差逆伝播法を適用することができないため、次数列 S_i のサンプリング処理後の次数列 S_{i+1} を近似した次数列 S'_{i+1} を求める方法を提案する。誤差逆伝播法とは、機械学習モデルの学習において、出力値と目標値の誤差を打ち消すように、各層の重みを調整する手法のことである。提案手法において、新規ノードと接続する既存ノードを m 個選ぶ際、同じノードを選ぶことはない。しかし、一度選択されたノードを選択しないとすると計算が煩雑になってしまうため、提案手法においては同じノードが選択されても良い場合の期待値を用いる。以上より S'_{i+1} は、以下の式で求められる。ただし、 S_i, S'_{i+1} は N 次の行ベクトルであり、 N は生成するグラフのノード数である。また、 m は定数である。

$$S'_{i+1} = S_i + mP^{\text{Modified}}. \quad (2)$$

ここで、次数列の要素として非負の実数を許容する拡張次数列という概念を導入する。式(2)で求められる次数列 S'_{i+1} の要素は一般に整数でないため、整数のみを要素とする次数列という言葉を用いるのは適切でない。そこで、本稿では、次数列の要素に整数のみを許容する場合は次数列、非負の実数を許容する場合は拡張次数列という言葉を用いることとする。

3.4 モデル構成

提案手法において、Graph Generator 部は LSTM 部、Enhancer 部という2つの機械学習ブロックと、確率分布 $P^{\text{BA}}, P^{\text{Modified}}$ 及び次の次数列の近似値 S'_i を求める3つの計算ブロックからなる。LSTM 部 (LSTM 層前の MLP 層を含む) は、次数列 S_i の時間発展を入力とし、それまでの次数列の情報を次のステップに伝えることを目的とする。最初の LSTM 部の入力とは j 番目の初期グラフ G_1^j を次数列 S_1 に変換したものに目的の特徴量 $F_j^{\text{Specified}}$ を加えたものであり、次のステップでは確率分布 P^{Modified} によって更新された次数列 S_2 及び $F_j^{\text{Specified}}$ を入力とする。この操作を指定の回数繰り返して、次数列を時間発展させることで、目的の特徴量を持つグラフを生成する。目的の特徴量 $F_j^{\text{Specified}}$ は、グラフ G_i^j に対してランダムに指定され、最終的に B 個のグラフを生成する。次数列の時間発展がグラフ構造に関する情報を持つという仮定が正しい場合、LSTM 部は次数列の時間発展から現在のグラフ構造を推定し、次のステップに伝えることができる。Enhancer 部は、目的の特徴量を持つグラフへ成長させるために、LSTM 部から出力される次数列の時間発展の情報を元にして、確率分布 P^{BA} を補正するための確率分布 P^{Enhancer} を出力する。Enhancer 部において、LSTM 部の出力はまず3層の MLP 層に入力される。その後、MLP 層の出力を Softmax 関数により確率分布に変換し、Enhancer として出力する。Softmax 関数とは、入力されたベクトル $\mathbf{x} = (x_1, x_2, \dots, x_J)$ の要素の総和が1になるように変換する関数であり、以下のよう定義される。ただし、 J は定数である。

$$\text{Softmax}(\mathbf{x})_i = \frac{\exp(x_i)}{\sum_{j=1}^J \exp(x_j)}. \quad (3)$$

計算ブロックは3つ存在し、1つ目は次数列 S_i から次数に比例した確率分布 P^{BA} を計算するブロックであり、以下の式で計算される。ただし、 $\sigma(S_i)$ は次数列 S_i の要素の総和を意味する。

$$p^{BA} = \frac{1}{\sigma(S_i)} S_i. \quad (4)$$

2つ目は、BA Model の確率分布 p^{BA} と Enhancer の確率分布 $p^{Enhancer}$ を用いて、Enhancer 付き確率分布 $p^{Modified}$ を計算するブロックであり、式 (1) で計算される。3つ目は、次数列 S_i と新規ノードと接続する既存ノードの個数 m を用いて、次の次数列の近似値 S'_i を計算するブロックであり、式 (2) で計算される。

Feature Estimator 部は、Graph Generator 部の出力である拡張次数列 S'_i の時間発展を次のステップに伝えるための LSTM 部 (LSTM 層前の MLP 層を含む) と、生成されたグラフの特徴量 $F^{Generated}$ を推定するための Feature Predictor (MLP 層) から構成される。Feature Estimator 部の目的は、Graph Generator 部が生成したグラフの特徴量を推定し、Graph Generator 部にフィードバックすることである。

4. 実験

4.1 実験概要

本実験の目的は、次の3点である。1つ目は、次数列及び拡張次数列によるグラフのシーケンス化という手法により、Graph Generator 部及び Feature Estimator 部がグラフ構造を学習できるかを検証し、このシーケンス化手法の有効性を確認することである。2つ目は、Graph Generator 部が Feature Estimator 部からのフィードバックによって、データセットを用いることなく、目標とする特徴量 $F_j^{Specified}$ を持つグラフを生成できるかを検証することである。3つ目は、生成されたグラフのべき指数が

BA Model の特性を概ね維持し、スケールフリー性を持つグラフを生成できるかを検証することである。

詳細な実験条件は以下の通りである。交互学習は、Graph Generator 部の学習と Feature Estimator 部の学習を1つのセットとし、メインループの回数 (100 回) 繰り返す。また、本実験における初期グラフにはただ1つの頂点とそれにつながる3つのノードを持つようなグラフを用いる。(図 2(b))

- グラフ数 B : 200
- ノード数 N : 300
- 新規ノードと接続する既存ノードの個数 m : 3
- 指定するクラスタ係数: [0.1, 0.2, 0.3]
- メインループ: 100
- Graph Generator 部の学習回数: 15
- Graph Generator 部の学習率: 0.001
- Feature Estimator 部の学習回数: 30
- Feature Estimator 部の学習率: 0.001

4.2 モデルのハイパーパラメータ

本節では、提案手法における Graph Generator 部と Feature Estimator 部のハイパーパラメータについて述べる。まず、2つのモデル共通のハイパーパラメータについて述べる。各モデル内で定義される3層の全結合層のドロップアウト率は全て 0.1 であり、活性化関数は ReLU 関数を用いる。次に、各モデル内で定義される LSTM 層の入力サイズは 128、隠れ層のサイズは 256 であり、ドロップアウト率は 0.1、レイヤーは 2 層である。また、最初の隠れ層の値は 0 で初期化される。

Graph Generator 部は、3層の全結合層、LSTM 層及び Enhancer

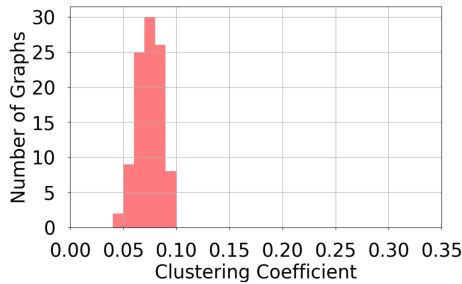


図4 BA Model から生成されたグラフのクラスタ係数のヒストグラム

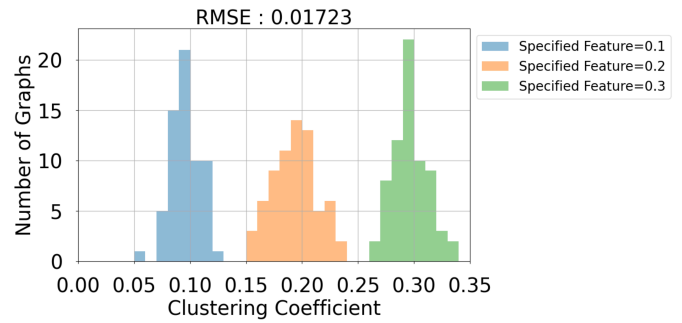


図6 提案手法によって生成されたグラフのクラスタ係数のヒストグラム

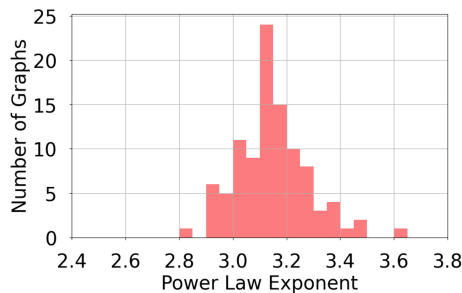


図5 BA Model から生成されたグラフのべき指数のヒストグラム

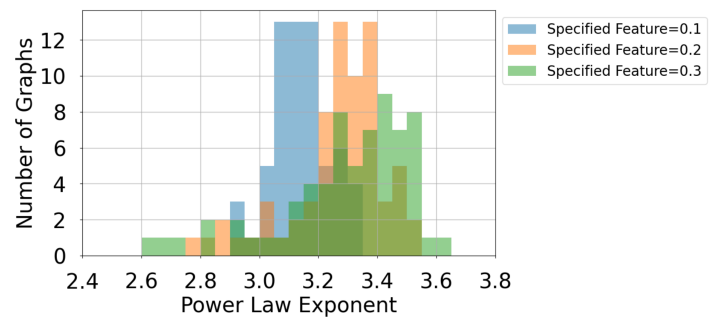


図7 提案手法によって生成されたグラフのべき指数のヒストグラム

部から構成される。最初の3層の全結合層では、サイズ100の次数列を1, 2層目で128, 3層目でLSTM層の入力サイズ(本実験では128)に変換し、LSTM層に入力する。また、最初の隠れ層の値は0で初期化される。Enhancer部は、3層の全結合層とSoftmax関数で構成され、LSTM部の隠れ層の出力を入力とし、1, 2層目で128, 3層目で次数列の次元数(本実験では100)に変換する。そして、3層目の出力をSoftmax関数に通し、確率分布を出力する。

Feature Estimator部は、3層の全結合層、LSTM層及びFeature Predictor部から構成される。最初の3層の全結合層では、サイズ100の拡張次数列を1層目で512, 2層目で256, 3層目でLSTM層の入力サイズ(本実験では128)に変換し、LSTM層に入力する。Feature Predictor部は、3層の全結合層で構成され、LSTM部の隠れ層の出力を入力とし、1層目で64, 2層目で32, 3層目で特徴量の次元数(本実験では1)に変換する。

4.3 評価方法

本実験では、Root Mean Squared Error (RMSE) とヒストグラムを用いてモデルの性能を評価する。RMSEは、生成されたグラフの特徴量 $F_j^{\text{Generated}}$ と指定した特徴量 $F_j^{\text{Specified}}$ の差を評価する指標であり、RMSEが小さいほど生成されたグラフの特徴量が指定した特徴量に近いことを示す。ヒストグラムは、生成されたグラフのクラスタ係数やべき指数の分布を示すことで、生成されたグラフがBA Modelの特性を維持しているかを評価する。本実験においてRMSEは、 j 番目に生成されたグラフ G_N^j の特徴量 $F_j^{\text{Generated}}$ と指定した特徴量 $F_j^{\text{Specified}}$ を用いて以下の式で定義される。ただし、 B は生成されるグラフの個数である。

$$\text{RMSE} = \sqrt{\frac{1}{B} \sum_{j=1}^B (F_j^{\text{Generated}} - F_j^{\text{Specified}})^2}. \quad (5)$$

4.4 性能評価

比較対象として、BA Modelを用いて生成したグラフのクラスタ係数とべき指数のヒストグラムを図4と図5に示す。図4より、BA Modelのクラスタ係数は概ね0.05から0.10の範囲に分布していることがわかる。図5より、BA Modelのべき指数は概ね2.85から3.7の範囲に分布していることがわかる。

図6より、指定したクラスタ係数[0.1, 0.2, 0.3]を持つグラフが生成されていることがわかる。また、このときのRMSEは0.017程度であった。また、クラスタ係数0.2や0.3はBA Modelによって生成されたグラフには存在しないため、提案手法は同条件のBA Modelでは生成されることのないグラフを生成する能力があることがわかった。図7より、提案手法によって生成されたグラフのべき指数は概ね2.6から3.8の範囲に分布していることがわかる。従って、元のBA Modelの特性を概ね維持しており、スケールフリー性を持つグラフを生成することができていることがわかる。

5. おわりに

本稿では、BA Modelを基盤としたDeep Learning手法を提案し、交互学習という手法を用いることでデータセットを用いる

ことなく、目標とする特徴量を持つグラフを生成する手法を提案し、その有効性を示した。実験においては、クラスタ係数を指定して学習及び生成を行い、RMSEがおよそ0.017という精度で目標とするクラスタ係数を持つグラフを生成することができた。また、ヒストグラムからも指定されたクラスタ係数のグラフが生成されたことが確認でき、提案手法によって生成されたグラフは、BA Modelの特性であるスケールフリー性を概ね維持していることも確認できた。本稿では、グラフのシーケンス化手法として次数列及び拡張次数列を用いることを提案したが、実験結果から、次数列はグラフ構造を表現するのに十分な情報を有しており、グラフのシーケンス化手法として有効であることを示した。

本稿での実験を踏まえ、提案手法にはいくつかの展望がある。まず、クラスタ係数以外の特徴量の指定精度に関する検証である。ランダムベースの手法においても、スケールフリー性とクラスタ係数を制御するモデルは存在するため[11]、それ以外の特徴量についても検証を行うことは提案手法の汎用性を評価する上で重要である。また、複数の特徴量の同時制御に関する検証も必要である。複数の特徴量の同時指定を行い、生成可能なグラフの境界を探ることは非常に興味深い課題である。

謝 辞

本研究の一部は科研費23H03379の助成を受けたものである。

文 献

- [1] P. Erdős and A. Rényi, "On Random Graphs I," *Publicationes Mathematicae*, vol.6, no.26, pp.290–297, 1959.
- [2] A.-L. Barabási and R. Albert, "Emergence of Scaling in Random Networks," *Science*, vol.286, no.5439, pp.509–512, 1999.
- [3] M. Simonovsky and N. Komodakis, "GraphVAE: Towards Generation of Small Graphs Using Variational Autoencoders," *Proceedings of the 27th International Conference on Artificial Neural Networks (ICANN 2018)*, pp.412–422, Rhodes, Greece, 2018.
- [4] J. You, R. Ying, X. Ren, W.L. Hamilton, and J. Leskovec, "GraphRNN: Generating Realistic Graphs with Deep Auto-regressive Models," *Proceedings of the 35th International Conference on Machine Learning (ICML 2018)*, pp.5694–5703, Stockholm, Sweden, 2018.
- [5] W. Jin, R. Barzilay, and T. Jaakkola, "Junction Tree Variational Autoencoder for Molecular Graph Generation," *Proceedings of the 35th International Conference on Machine Learning (ICML 2018)*, pp.2328–2337, Stockholm, Sweden, 2018.
- [6] Y. Du, X. Guo, A. Shehu, and L. Zhao, "Interpretable Molecular Graph Generation via Monotonic Constraints," *Proceedings of Proceedings of the 2022 SIAM International Conference on Data Mining (SDM)*, pp.73–81, Minneapolis-St. Paul Twin Cities, MN, USA, 2022.
- [7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Nets," *Proceedings of the 27th International Conference on Neural Information Processing Systems (NIPS 2014)*, Montreal, QC, Canada, 2014.
- [8] X. Guo and L. Zhao, "A Systematic Survey on Deep Generative Models for Graph Generation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.45, no.5, pp.5370–5390, 2023.
- [9] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol.9, no.8, pp.1735–1780, 1997.
- [10] D.P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," *Proceedings of the 2nd International Conference on Learning Representations (ICLR 2014)*, Banff, Canada, 2014.
- [11] P. Holme and B.J. Kim, "Growing Scale-Free Networks with Tunable Clustering," *Physical Review E*, vol.65 2 Pt 2, p.026107, 2001.