深層生成モデルによる時系列ネットワークの低次元埋め込み

Embedding time-dependent networks into low-dimensional space using deep generative models

円道 滉一郎 [1]* 江口 浩二 [1] 羽森 茂之 [2] 金京 拓司 [2]
Koichiro Endo[1] Koji Eguchi[1] Shigeyuki Hamori[2] Takuji Kinkyo[2]

[1] 神戸大学大学院システム情報学研究科
[1] Graduate School of System Informatics, Kobe University
[2] 神戸大学大学院経済学研究科
[2] Graduate School of Economics, Kobe University

Abstract: Network embedding is one of the approaches to effectively analyzinge the network data. Almost all the existing network embedding methods adopt *shallow* models without having deep architecture that is commonly used in deep learning studies. However, shallow models cannot capture highly non-linear network structures that are often observed in real-world, complex networks. To solve this problem, Structural Deep Network Embedding (SDNE) was proposed as a *deep* model for network embedding. In this paper, we focus on Generative Stochastic Network (GSN) for network embedding, in an extension of Autoencoder. GSN robustly captures latent features of data by adding random noises in the process of learning. The framework to capture the latent structure of network is similar to that of SDNE. As a target network in this study, we focus on the time-dependent networks. In order to address the dependency between time intervals and to capture the tendency of previous time interval, we propose *time-dependent pretraining* that uses the parameters learned from the previous time interval as initial states of the current time interval while in the learning process. In the experiments, we use time-dependent financial network data, where each node (or vertex) represents a bank and each link (or directed edge) represents a per-month transaction between a pair of banks, resulting in a series of per-month networks.

1 はじめに

現実世界には，人間関係，企業間の関係などといったノードとリンクで構成されるネットワークで表現できる関係データが数多く存在し，ネットワーク表現を分析することで，コミュニティの検出や未観測の関係性の有無の予測などを行うことができる．その分析における有効的な方法の１つとして，ネットワークを低次元潜在空間に埋め込むアプローチ (以下,「ネットワーク埋め込み」と呼ぶ) がある．今日まで様々なネットワーク埋め込み手法が提案されてきたが，そのほとんどは浅い構造を持つモデルであり，複雑である潜在的なネットワーク構造を十分に捉えることが困難であった．そのような中，自己符号化器 [1] と呼ばれるニューラルネットワークを用いた次元圧縮を行うモデルを深い構造にした深層自己符号化器 [1] に基づく構造的深層ネットワーク埋め込み (Structural Deep Network Embedding: SDNE)[2] と呼ばれるモデルが提案された．

本論文では，以上に述べた動向を踏まえた上で，時系列のネットワークに着目する．とりわけ，ネットワークの時間変化を分析する際に時間的依存性を反映させるようなネットワーク埋め込み手法について提案する．時系列のネットワークの中でも，銀行間の取引を表した金融データに着目し，月ごとの取引をネットワークで表現した時系列のネットワークデータを用いて分析を行った．そして，ネットワークの構造が保存されているかどうかの検証と過去の取引の動向を利用することによる未知の関係性の予測を行った．また，ネットワークのノード間の関係性は数量的に表すこともできる．そこで，本論文では，金融データを月ごとの取引金額を重みとした重み付きネットワークでも表現した．そして，手法により重みの大小関係を捉えられているかの検証を行い，未観測リンクに関する重みの大小関係の予測を行った．

*連絡先：神戸大学大学院システム情報学研究科
〒 657-0013 兵庫県神戸市灘区六甲台町 1-1
E-mail: endo@cs25.scitec.kobe-u.ac.jp

2 関連研究

ネットワーク埋め込みにおいて，ネットワークの複雑な構造をどのように捉えるかという方法を考えるのは重要な課題である．ネットワークの複雑な構造を捉えるモデルとして，構造的深層ネットワーク埋め込み (Structural Deep Network Embedding: SDNE) というモデルが提案され，良い性能であったことが文献 [2] に述べられている．本研究でネットワーク埋め込みを行う際に参考にした SDNE を紹介するにあたり，まずその基本となる自己符号化器について述べる．次にネットワーク埋め込みの際に考慮する問題点を述べてから SDNE の詳細について述べる．本研究におけるネットワーク埋め込みにおいても自己符号化器に着目しており，ネットワーク埋め込みに使用した自己符号化器の拡張モデルも加えて紹介する．

2.1 自己符号化器

自己符号化器 (Autoencoder: AE)[1] はニューラルネットワークを用いて次元圧縮を行い，データの特徴表現を得るためのモデルである．モデルの構成は入力層，中間層，出力層の 3 層構造のニューラルネットワークを成しており，入力層と出力層の次元数を同じにして，出力されたデータが入力されたデータを復元するように学習させるものである．自己符号化器の構造の例を図 1 に示す．

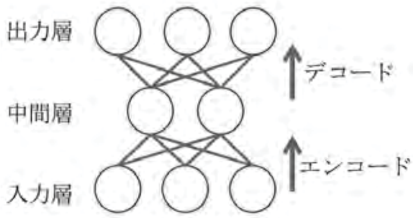

図 1: 自己符号化器の構造.

中間層の次元数は恒等写像の学習にならないように，データの次元数より小さくする必要がある．入力層 $\mathbf{X}$ に対して，中間層 $\mathbf{Y}$ は $\mathbf{Y} = f_\theta(\mathbf{X}) = \phi(\mathbf{WX}+\mathbf{b})$ と定義され，出力層 $\mathbf{X}'$ は $\mathbf{X}' = f'_{\theta'}(\mathbf{Y}) = \phi'(\mathbf{W}'\mathbf{Y}+\mathbf{b}')$ と定義される．ここで，パラメータ θ, θ' はそれぞれ $\theta = (\mathbf{W}, \mathbf{b}), \theta' = (\mathbf{W}', \mathbf{b}')$ であり，$\mathbf{W}, \mathbf{W}'$ は重み，$\mathbf{b}, \mathbf{b}'$ はバイアスである．また，ϕ, ϕ' は活性化関数と呼ばれ，活性化関数にはシグモイド関数やランプ関数 (ReLU 関数) などが用いられる．活性化関数とパラメータを用いて入力層から中間層の表現を得ることをエンコード，中間層の表現から出力層の表現を得ることをデコードと呼ぶ．$\mathbf{W}'$ に $\mathbf{W}$ の転置行列を使う考え方があり，その考え方は”tied weight”と呼ばれるが，”tied weight”の場合は学習で考慮するパラメータが実質 $\mathbf{W}, \mathbf{b}, \mathbf{b}'$ となる．

このように定義された出力層 $\mathbf{X}'$ を入力層 $\mathbf{X}$ に近づけるようにパラメータを学習する．具体的には，損失関数を L とすると，$L(\mathbf{X}, \mathbf{X}')$ の値の最小化することが学習となる．損失関数には 2 乗誤差や交差エントロピーなどがよく用いられ，損失関数の最小化には確率勾配降下法 [4] などが用いられる．

学習が終了し，得られたパラメータを用いて表される中間層の表現が次元圧縮された入力データの特徴表現である．

2.2 生成的確率ネットワーク

自己符号化器の一種に雑音除去自己符号化器 (Denoising Autoencoder: DAE)[5] というモデルがある．基本的な学習の考え方は自己符号化器と同様であるが，違う点は，自己符号化器が $L(\mathbf{X}, \mathbf{X}')$ の最小化を考えるに対して，DAE は入力データ $\mathbf{X}$ にノイズを加えた表現 $\tilde{\mathbf{X}}$ にエンコードとデコードを行い得られた表現 $\tilde{\mathbf{X}}'$ を用いて，$L(\mathbf{X}, \tilde{\mathbf{X}}')$ の値の最小化を考えることである．そして，DAE を一般的な確率モデルとして捉え直し，以下に示した $\mathbf{X}$ と $\tilde{\mathbf{X}}$ を交互に生成するマルコフ連鎖を考える．図示すると図 2 のようになる．

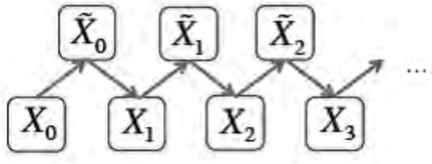

図 2: 雑音除去自己符号化器のマルコフ連鎖.

このマルコフ連鎖において，$\mathbf{X}_0$ が入力データ $\mathbf{X}$ に対応し，$\mathbf{X}_1$ が入力データ $\mathbf{X}$ を再構築したものに対応する．マルコフ連鎖において，サンプリングを繰り返していくと誤ったモードを推定してしまう可能性がある．それを回避するため，マルコフ連鎖のサンプリングの過程で得られた $\tilde{\mathbf{X}}_t$ も訓練データとして用いることによって誤ったモードを元のモードに引き戻すことを行う．これを Walkback アルゴリズムと呼ぶ．この Walkback アルゴリズムを適用した DAE[6] も提案されている．そして，Walkback アルゴリズムを適用した DAE を拡張したモデルに生成的確率ネットワーク (Generative Stochastic Network: GSN)[7] がある．より柔軟で，複雑な構造を持つことができるように DAE に潜在層を加えたものが GSN である．例として潜在層を 2 層持つ GSN のマルコフ連鎖を図 3 で示す．

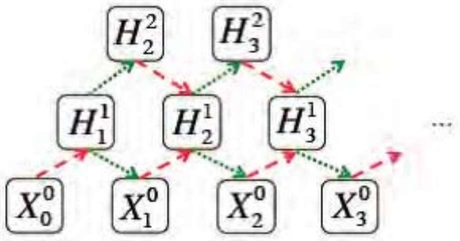

図 3: 生成的確率ネットワークのマルコフ連鎖の例.

GSN の入力層の表現は 1 層目の潜在層によって更新され，最上層以外の潜在層は上下の 2 層によって更新され，最上層は 1 つ下の潜在層によって更新される．上図の各層の表現は次のように示される．

$$\mathbf{X}_t^0 = \phi(\mathbf{W}_1^T \mathbf{H}_t^1 + \mathbf{b}_0) \tag{1}$$
$$\mathbf{H}_t^1 = \phi(\mathbf{W}_1 \mathbf{X}_{t-1}^0 + \mathbf{W}_2^T \mathbf{H}_t^2 + \mathbf{b}_1) \tag{2}$$
$$\mathbf{H}_t^2 = \phi(\mathbf{W}_2 \mathbf{H}_{t-1}^1 + \mathbf{b}_2) \tag{3}$$

ここで，$\mathbf{W}_1, \mathbf{W}_2$ は重み，$\mathbf{b}_0, \mathbf{b}_1, \mathbf{b}_2$ はバイアスである．GSN の学習では Walkback アルゴリズムが適用され，サンプリング回数が T 回の時のモデルの構造は図のマルコフ連鎖を T 回で打ち切ったものとなる．DAE 同様，データにノイズを加えた表現を GSN に入力し，再構築されたものが元のデータに近づくようにパラメータを学習する．GSN の学習では奇数目の層と偶数目の層を交互にサンプリングして更新する．そして，再構築された表現にまたノイズを加える．また，潜在層の表現は活性化関数に代入する前後にノイズを加える [8]．本研究では，GSN をネットワーク埋め込みに適用する．

2.3 ネットワーク埋め込みに関する問題定義

ネットワーク表現を学習するにおいて，3 つの問題点がある．それは，高度非線形性，構造の保存，スパース性である．まず，高度非線形性についてであるが，高度非線形なネットワーク構造をよく捉えるために，Structural Deep Network Embedding(SDNE)[2] と呼ばれるディープラーニングの手法を用いた深層モデルが提案された．本研究では，ネットワーク埋め込みを行う際に SDNE の考え方を用いている．SDNE の詳細については，次節で述べる．そして，構造の保存やスパース性の問題に対処するため，1 次近接性と 2 次近接性をネットワーク埋め込みを行う際に取り入れる必要がある．1 次近接性は局所的な構造に着目したもので，ノード間にリンクを持つならば，それらのノードは似ているという考え方を仮定している．ある論文はその論文が引用している論文といくつかの共通した話題が含まれている可能性があるというようなことが例として挙げられる．2 次近接性は全域的な構造に着目したもので，同じような隣接構造をもつノード同士は似ているという考え方を仮定している．多くの共通の友達がいる人同士は友達である可能性があるといったようなことが例として挙げられる．現実世界に存在するネットワークはリンクが欠損している場合もあり，2 次近接性はリンクが欠損していたとしてもノードの近接性を捉えることができるので，ノードの関係性をより豊富な情報を用いて捉えることができる．これらの近接性を考慮することにより，局所的にも全域的にもネットワークの構造を捉え，多い情報を用いてネットワークを特徴づけることができるので，ネットワークの構造の保存やスパース性の問題を緩和できる．1 次近接性はノード間のリンクの有無のみに着目しているので，ネットワークが無向グラフの場合にのみ有効であることに注意が必要である．

2.4 構造的深層ネットワーク埋め込み

本節では，構造的深層ネットワーク埋め込み (Structural Deep Network Embedding: SDNE)[2] について述べる．SDNE は自己符号化器を深層にした深層自己符号化器に基づき，枠組みは図 4 のようになる．

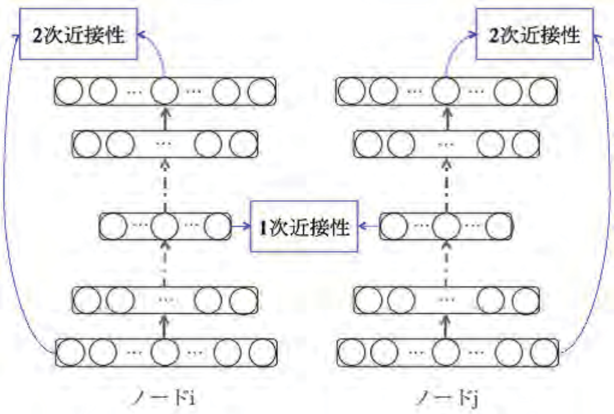

図 4: 構造的深層ネットワーク埋め込みの枠組み.

ネットワークデータは隣接行列で表現され，それを入力とする．隣接行列の i 行 j 列要素を $s_{i,j}$，ネットワークデータのノード数を N，SDNE における入力層の表現を $\mathbf{X}$，出力層の表現を $\tilde{\mathbf{X}}$，最も低次元な層の表現を $\mathbf{Y}$ とする．まず，1 次近接性からノード間にリンクを持つなら $\mathbf{Y}$ も似たようであると考えられ，1 次近接性の損失関数 L_{1st} は以下のように示される．

$$L_{1st} = \sum_{i,j=1}^{N} s_{i,j} ||\mathbf{y}_i - \mathbf{y}_j||_2^2 \tag{4}$$

次に 2 次近接性から似た隣接構造を持つノードは似たような再構築表現になると考えられ，2 次近接性の損失関数 L_{2nd} は以下のように示される．

$$L_{2nd} = ||(\tilde{\mathbf{X}} - \mathbf{X}) \odot \mathbf{B}||_F^2 \tag{5}$$

ここで，$\odot$ はアダマール積であり，$||\cdot||_F$ はフロベニウスノルムである．アダマール積とは同じサイズの行列に対して行列の要素ごとに積を取ることによって定まる演算であり，フロベニウスノルムは行列 A の i 行 j 列要素を $a_{i,j}$ とすると，$||A||_F = \sqrt{\sum_{i,j} a_{i,j}^2}$ で定義される．そして，$\mathbf{B}$ はバイアスであり，$\mathbf{B}$ によって，ネットワークのスパース性に対処する．$\mathbf{B}$ の i 行 j 列要素を $b_{i,j}$ とすると，もし $s_{i,j} = 0$ ならば $b_{i,j} = 1$，そうでなければ $b_{i,j} = \beta > 1$ とする．ここで，β はパラメータである．再構築誤差を最小化することで，再構築の手順で似た隣接構造を持つノードは似た潜在表現を持つようになる．

以上の 2 つの損失関数を組み合わせ，さらに過学習を抑制するための正則化項 L_{reg} を加えたものを，SDNE の目的関数とする．目的関数 L は次のように表現される．

$$L = \alpha L_{1st} + L_{2nd} + \gamma L_{reg} \tag{6}$$

ここで，α, γ はパラメータである．目的関数を最小化することにより，1 次近接性と 2 次近接性を考慮したネットワーク埋め込みを行うことができる．尚，1 次近接性は無向グラフの場合のみ有効であるので，有向グラフの場合はパラメータ α を $\alpha = 0$ とする必要がある．

3　時系列ネットワーク埋め込み

本研究では，構造的深層ネットワーク埋め込み (Structural Deep Network Embedding: SDNE) が深層自己符号化器に基づくことに着目し，同じ自己符号化器の系統の生成的確率ネットワーク (Generative Stochastic Network: GSN) をネットワーク埋め込みに用いることを考える．尚，GSN でネットワーク埋め込みを行う際，ネットワークデータの構造を捉える枠組みは SDNE と同様のものを使用する．本章では，まず GSN を用いたネットワーク埋め込みに関する詳細について述べ，次に，GSN で時系列ネットワークデータの埋め込みをどのように行うかを提案する．

3.1　生成的確率ネットワークにおけるネットワークデータの適用

SDNE のネットワーク埋め込みの枠組みを参考にし，GSN では最上層で 1 次近接性，最下層で 2 次近接性を考慮する．ネットワーク埋め込みにおける GSN の 1 次近接性と 2 次近接性を捉えた目的関数は次の式で表すことができる．尚，ネットワークデータのノード数を N，隣接行列を $S = \{s_{i,j}\}_{i,j=1}^N$，GSN の潜在層の数を K，入力層の表現を $\mathbf{X}_t^k = \{\mathbf{x}_{i_t}^0\}_{i=1}^N$，最上層の表現を $\mathbf{H}_t^K = \{\mathbf{h}_{i_t}^K\}_{i=1}^N$，Walkback 回数を T 回とする．

$$\begin{aligned} L &= \frac{\alpha}{T-K+1}\sum_{t=K}^{T}(L_{1st})_t + \frac{1}{T}\sum_{t=1}^{T}(L_{2nd})_t + \gamma L_{reg} \\ &= \frac{\alpha}{T-K+1}\sum_{t=K}^{T}(\sum_{i,j=1}^{N} s_{i,j}||\mathbf{h}_{i_t}^K - \mathbf{h}_{j_t}^K||_2^2) \\ &\quad + \frac{1}{T}\sum_{t=1}^{T}||(\tilde{\mathbf{X}}_\mathbf{t}^\mathbf{0} - \mathbf{X}_t^0) \odot \mathbf{B}||_F^2 + \gamma L_{reg} \end{aligned} \tag{7}$$

SDNE と同様，α, γ はパラメータであり，$(L_{1st})_t, (L_{2nd})_t$, L_{reg} は，それぞれ Walkback が t 回目の 1 次近接性の損失関数，Walkback が t 回目の 2 次近接性の損失関数，重みを用いた正則化項である．そして，$\mathbf{B}$ はバイアスであり，$\mathbf{B}$ の i 行 j 列要素を $b_{i,j}$ とすると，もし $s_{i,j} = 0$ ならば $b_{i,j} = 1$，そうでなければ $b_{i,j} = \beta > 1$ とする．しかし，本研究において時系列ネットワークデータを扱う上での例外として，もしある時点において $\mathbf{s}_i = \{s_{i,j}\}_{j=1}^N = \mathbf{0}$ である．すなわち，どのノードともリンクを持たないならば，ノード i を欠損扱いとし，$\mathbf{b}_i = \{b_{i,j}\}_{j=1}^N = \mathbf{0}$ とする．このように工夫する事で学習の際に欠損ノードが直接影響を及ぼすことがなくなる．ネットワークが無向グラフの時はパラメータ α は $\alpha = 0$ となる．GSN におけるネットワーク埋め込みは，ネットワークデータを隣接行列で表現したものを入力とし，上式の最小化を行うことによってネットワークの潜在構造を捉え，保存することを目的とする．

3.2　生成的確率ネットワークにおける時系列ネットワークデータの低次元埋め込み

本節では，前節で述べたネットワーク埋め込みを行う GSN を用いて，時系列のネットワーク埋め込みを考える．本研究では，ネットワークデータの傾向や特徴を捉えるためのパラメータに着目し，時系列を考慮する．尚，ここで述べているパラメータとは，層間の関係を表す重みと各層のバイアスのことを示している．そして，本研究では，事前に学習したパラメータを初期値とするという点に着目した．過去のデータの学習によって得られたパラメータを現在のデータの学習における初期値とする，ある種の事前学習 (プレトレーニング) を行うことで時区間の間の依存性を捉え，過去の動向を踏まえた学習ができるのではないかと考え，時系列ネットワーク埋め込み手法として提案する．

例として，本研究で着目する時系列ネットワークデータである金融データを用いる場合を考える．1 か月間の企業間のやり取りの関係を示す時系列ネットワークデータに時系列プレトレーニングを適用する時，当月のデータを用いて学習する際にパラメータの初期値を先月のデータを用いて学習して得た最適パラメータとする．そして，翌月のデータに対して学習を行う際に，先月の学習で得たパラメータを初期値とした当月の学習によって得られたパラメータを初期値とすれば，先月と当月の傾向をある程度考慮した状態で翌月のデータを学習することができる可能性が考えられる．

4 実験

時系列ネットワーク埋め込みを GSN で行い，性能評価実験を行う．本研究では，時系列ネットワークを重みなしと重み付きの 2 種類の隣接行列で表したデータとして使用する．この章において，時系列で過去のデータの傾向を考慮し学習する事を時系列プレトレーニングと呼ぶ．

4.1 データセット

本研究では時系列ネットワークの中でも金融データを使用する．実験で用いた金融データのデータセットはリーマンショックが起こった 2008 年の銀行間の取引のデータである．欧州 15 か国の銀行間の取引を 1 年分集めたデータであり，銀行は全部で 177 行，取引数は 93067 である．1 つの取引において実験で着目する情報は，取引の契約日，取引を持ちかけられる側の銀行名 (quoter)，持ちかける側の銀行名 (aggressor)，最終的な取引金額，取引の状態である．最終的な取引金額は百万ユーロ単位で示されており，取引の状態は"Sell"か"Buy"の 2 つで示されている．そして，この 1 年分の銀行間の取引関係のネットワークデータを有向グラフと捉え，177 × 177 の重みなし隣接行列と重み付き隣接行列として月ごとに表現する．ここで，隣接行列を S とする時，S の i 行 j 列の要素を $s_{i,j}$ で表すとする．有向グラフの向きは取引の状態 ("Sell"or"Buy") で決めており，取引を持ちかけたかどうかで向きを決めていないことに注意が必要である．S をある月の重みなし隣接行列と考える場合，取引の有無で隣接行列を作成するため，$s_{i,j}$ は 0 か 1 の値を持つ．そして，S をある月の重み付き隣接行列と考える場合，最終的に成立した取引額がいくらであったかという数値を契約が成立した取引の規模と考え，その数値で隣接行列を作成するため，$s_{i,j}$ は月ごとの取引の規模の合計の値となる．重みなしと重み付き隣接行列のリンク密度の月ごとの平均は約 0.0585 である．元の隣接行列のリンク数の月ごとの平均は 1833.25 であり，欠損ノード数の月ごとの平均は 25.25 である．ここで，欠損ノード数は，どのノードともリンクを持たないノードの数であり，GSN の学習において，欠損ノードは目的関数に影響を与えないよう工夫している．

4.2 重みなし隣接行列を用いた実験

この節では，重みなし隣接行列を GSN の入力とした時の性能を隣接行列の再構築と未観測のリンク予測というタスクで評価する．実験を行う際，元の隣接行列のリンクをランダムに 15%除いた隣接行列を使用する．Walkback 回数 T は $T = 5$，$(\alpha, \beta, \gamma) = (0, 10, 10^{-6})$ とする．これらのパラメータの値は，本実験で使用するデータを扱う際に適切になるよう予備実験により定めた．そして，本実験で使用する GSN は入力層が 177 次元，1 層目の潜在層が 100 次元，2 層目の潜在層が 50 次元の 3 層構造である．

4.2.1 隣接行列の再構築に関する評価

再構築の評価はネットワーク埋め込みにおいて最も基本的な評価である．最後の Walkback で再構築された隣接行列の各要素の値に着目し，リンクを持つ可能性の高い順番でランキングを行い，実際のリンクをどの程度適合させることがを平均適合率 (Average Precisioon) で評価する．評価には時系列プレトレーニングの有無による月ごと平均適合率の差を図 5 で示し，性能を比較する．1 月より過去のデータが存在しないため，1 月に関しては時系列プレトレーニングが行うことができず，時系列プレトレーニングの有無によって数値に変動がないことに注意が必要である．

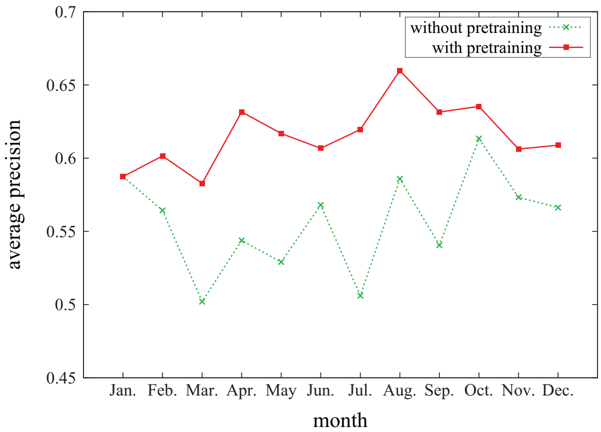

図 5: 重みなし隣接行列再構築における平均適合率．

月ごとの平均適合率に着目すると，すべての月において時系列プレトレーニングを行った方が性能は良く，改善率 (パーセント) の月ごとの平均が 11.9%で，その標本標準偏差が 5.8%であった (以下，月ごとの性能に着目した改善率を表記する時は「11.9±5.8%」のようにする)．また，月ごとの改善率 I は評価値 E から E' になった時，$I = E'/E \times 100 - 100$ と定義される．

4.2.2 未観測リンクの予測に関する評価

リンク予測は過去の傾向から現在の観測されていないリンクをどの程度予測できるかを評価するものである．あらかじめ除いた 15%のリンクを未観測データ，残りのリンクを観測データとする．そして，先月のパラメータを用いて当月のデータを再構築し，再構築された隣接行列において，観測されているリンクを持つノード対を除外して残りのノード対においてリンクを持つ可能性の高い順番でランキングを行い，未観測データのリンクをどれだけ捉えられるかを評価する．スパース性が大きく増すので，問題としての難易度は高くなる．リンク予測における時系列プレトレーニングは先々月のパラメータを初期値として学習した先月のパラメータを用いて当月を予測することである．時系列プレトレーニングの有無による月ごと平均適合率の差を図 6 で示し，性能を比較する．1 月より過去のデータが存在しないため，2 月に関しては時系列プレトレーニングが行うことができず，時系列プレトレーニングの有無によって数値に変動がないことに注意が必要である．

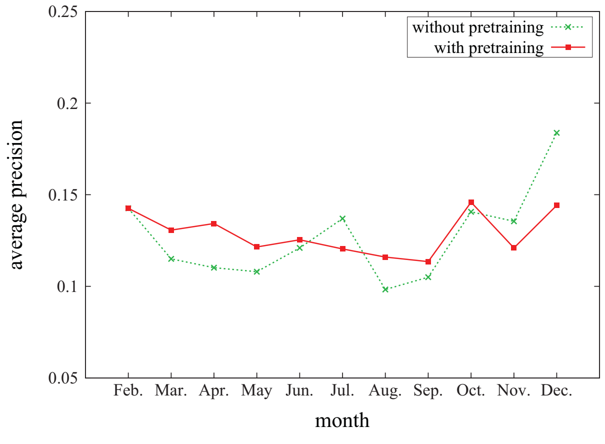

図 6: 重みなし隣接行列のリンク予測における平均適合率．

月ごとの平均適合率に着目すると，7 月，11 月，12 月以外は時系列プレトレーニングを行った方が平均適合率の値は良く，時系列プレトレーニングの有効性が部分的に確認された．

4.3 重み付き隣接行列を用いた実験

この節では，重み付き隣接行列を GSN の入力とした時の性能を隣接行列の再構築と未観測の重みの予測というタスクで評価する．2 次近接性を捉える際に関与するパラメータ β は $\beta = 40$ と設定している．そして，重みなし隣接行列を用いた時と同様に，入力層が 177 次元，1 層目の潜在層が 100 次元，2 層目の潜在層が 50 次元の 3 層構造の GSN を使用する．

4.3.1 隣接行列の再構築に関する評価

重み付き隣接行列を用いた時のネットワーク埋め込みの基本的な評価として，隣接行列の再構築を検討する．重み付き隣接行列の再構築では，リンクを持つノード対のみに着目する．そして，重みの重い順にノード対をランキングし，そのランキングが GSN の学習によって再構築されているかをケンドール順位相関係数を用いて評価する．ケンドール順位相関係数 τ は $|\tau| \leq 1$ であり，分布などを仮定せず，ランキング同士の相関度合いを測る評価指標である．このタスクにより取引の規模の大小関係を捉えているかを判断する．結果を月ごとに図 7 で示す．1 月より過去のデータが存在しないため，1 月に関しては時系列プレトレーニングが行うことができず，時系列プレトレーニングの有無によって数値に変動がないことに注意が必要である．

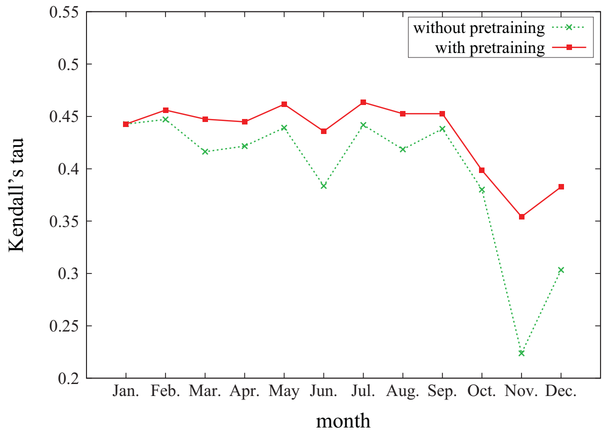

図 7: 重み付き隣接行列再構築におけるケンドール順位相関係数．

ここで,「ケンドール順位相関係数が 0 である (無相関である)」という仮説を帰無仮説とした有意確率を求めた時，いずれの月でも有意水準 1%よりも下回り，帰無仮説が棄却されることが確認された．したがって，いずれの月についても正解となる未観測の重みのランキングと再構築された隣接行列に基づく未観測の重みのランキングに統計的に有意な相関があると言うことが

でき，提案手法の有効性を示すことができた．時系列プレトレーニングを行った時のケンドール順位相関係数の改善率は約 12.7±15.8%であり，時系列プレトレーニングの有効性が確認できた．ただし，いずれの月も 2.0〜58.3%の正の改善率を示しており，標本標準偏差が大きいのは改善の程度が月によって幅があることを意味することに注意が必要である．

4.3.2　未観測重みの予測に関する評価

未観測の重みの予測は過去の傾向から現在の未観測リンクの重みの大小関係をどの程度予測できるかを評価するものである．あらかじめ除いた 15%のリンクを未観測データ，残りのリンクを観測データとする．まず，過去のデータを用いて学習されたパラメータを用いて現在の観測データを再構築する．そして，未観測データでリンクを持つノード対のみに着目し，観測データを再構築して得られた隣接行列の各要素から重みのランキングを行う．そして，そのランキングが未観測データのリンクの重みのランキングとどの程度相関があるかをケンドール順位相関係数を用いて評価する．時系列プレトレーニングを行う時は，先々月までの傾向を考慮し学習されたパラメータを先月の傾向を学習する際の初期値とし，学習されたパラメータを用いて当月の観測データを再構築したものから当月の未観測データのランキングを予測する．時系列プレトレーニングの有無による月ごとケンドール順位相関係数の変化を図 8 で示す．隣接行列の再構築のタスクの時と同様に，1 月より過去のデータが存在しないため，2 月に関しては時系列プレトレーニングが行うことができず，時系列プレトレーニングの有無によって数値に変動がないことに注意が必要である．

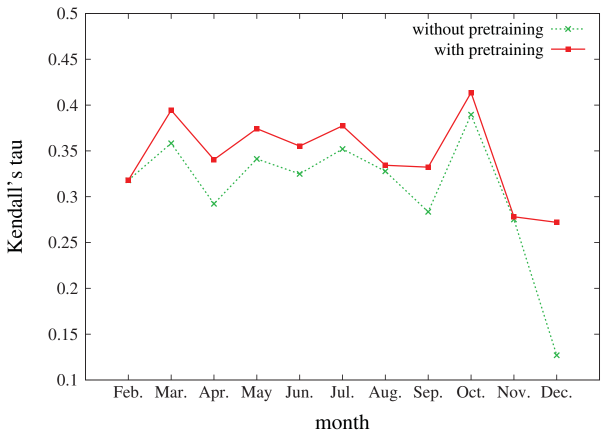

図 8: 重み付き隣接行列の重み予測におけるケンドール順位相関係数.

ここで，隣接行列の再構築の時と同様に「ケンドール順位相関係数が 0 である (無相関である)」という仮説を帰無仮説とした時，いずれの月でも帰無仮説が有意水準 1%の下で棄却されることが確認されたので統計的に有意な相関があると言うことができ，提案手法の有効性を示すことができた．全ての月において，時系列プレトレーニングを行うことによる性能改善が見られ，ケンドール順位相関係数の改善率は約 19.3±32.0%であった．以上の結果より，時系列プレトレーニングの有効性が確認できた．

4.4　考察

この節では，実験の結果について考察する．重みなし隣接行列の再構築では，時系列プレトレーニングを行うことで性能改善を確認でき，時系列プレトレーニングは有効であると言える．リンク予測では，時系列プレトレーニングにより，7 月，11 月，12 月以外は性能の改善が見られた．この 3 つの月で性能の改善が見られなかった原因として，ランダムに除かれた 15%のリンクの中に時系列の傾向を捉える際に重要なリンクが含まれていた可能性があることや取引があった銀行が月ごとに局所的であったことなどが挙げられる．平均適合率の値自体が高くない原因については，観測されたリンクを除くことでスパース性が増し，リンク予測がかなり困難なタスクになったためであると考えられる．重み付き隣接行列の再構築と未観測の重みの予測では，時系列プレトレーニングを行うことでケンドール順位相関係数の値は統計的に有意に大きな値を示していることから，時系列プレトレーニングは有効であると言える．

さらに，重み付き隣接行列は取引金額の値を利用していることから，取引金額と金利の結びつきがあると考え，時系列プレトレーニングを行った時の GSN の最上層の潜在表現に金利と相関があるかどうかを追加的に分析した．まず，銀行間の取引における金利の期待値を月ごとに求め，それを各要素に持つ金利期待値隣接行列を考える．そして，金利期待値隣接行列の列ごとに金利の期待値の平均を取り，それを銀行ごとの金利ラベルとする．この金利ラベルは，各銀行が他行から平均的にどの程度の金利で融資を受けたかを示し，金利ラベルの値が大きいとき，その銀行の経営が悪化していることの指標となり得る．GSN の最上層は 50 次元であり，それが銀行 177 行分だけ存在するので，177×50 の行列を得ることができる．その行列の列に着目し，列ベクトル (177 次元) が金利ラベルベクトル (177 次元) とどの程度相関があるかを調べる．このようにすることで，50 次元の最上層の中で金利を捉えるようと機能している潜在表現を確認できる可能性がある．相関を

調べる際，ピアソンの相関係数を用いた．ピアソンの相関係数を r とすると，$|r| \leq 1$ である．そして，金利との相関をピアソンの相関係数を用いて調べたところ，一部の潜在表現に金利との有意な相関が認められた．例として，4 月では $r = -0.1959, 0.3298, -0.1840$ となる 3 つ，9 月では $r = -0.2450$ となる 1 つ，10 月では $r = -0.2175, -0.2821$ となる 2 つの潜在表現が確認された．ここで，いずれの月においても「ピアソンの相関係数が 0 である (無相関である)」という帰無仮説の下で，有意確率は有意水準 5%よりも下回り，帰無仮説が棄却されることが確認されたので，統計的に有意な相関があると言うことができる．上記の通り，金利と相関のある潜在表現の数と基底が月により変動があった．9 月にリーマンショックが起こったことから，金利が非定常な動きをしていたことが予想され，潜在表現との相関も他の月と比して有意なものが観察されにくい状況が認められる．潜在表現と金利の関係についてより詳細な分析を行うことは今後の課題の一つである．

5 おわりに

銀行間の取引のデータに対し，重みなし隣接行列における隣接行列の再構築とリンク予測，重み付き隣接行列における隣接行列の再構築と未観測の重みの予測の 4 つのタスクで評価することで，GSN でネットワーク埋め込みを行うことの有用性と時系列プレトレーニングの有効性について確認することができた．

本研究における今後の課題として，まず，本実験で用いた銀行の取引データに関して，取引の有無や金額だけでなく，取引にかかった期間や取引の頻度に着目してネットワーク表現を獲得し，解析することが挙げられる．さらに，取引における金利まで考慮し，金利の予測を行うことができれば，より解析の幅は広がる．また，ネットワーク埋め込みを行うモデルは様々なので，それらのモデルを用いた時系列ネットワーク埋め込みの性能と比較を行う必要があると考えられる．

謝辞

本研究を行うにあたり，有益な助言，協力を頂いた神戸大学大学院システム情報学研究科谷口隆晴准教授，同大学経済学研究科の田中克幸助教に感謝する．本研究の一部は科学研究費補助金基盤研究 (B)(15H02703) の援助による．

参考文献

[1] Hinton, Geoffrey E., and Ruslan R. Salakhutdinov. ”Reducing the dimensionality of data with neural networks.” science 313.5786 (2006): 504-507.

[2] Daixin Wang, Peng Cui, Wenwu Zhu. ”Structural Deep Network Embedding.” KDD Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(2016): 1225-1234.

[3] Bengio, Yoshua. ”Learning deep architectures for AI.” Foundations and trendsR in Machine Learning 2.1 (2009): 1-127.

[4] Ngiam, Jiquan, et al. ”On optimization methods for deep learning.” Proceedings of the 28th International Conference on Machine Learning (ICML-11). 2011.

[5] Vincent, Pascal, et al. ”Extracting and composing robust features with denoising autoencoders.” Proceedings of the 25th international conference on Machine learning. ACM, 2008.

[6] Bengio, Yoshua, et al. ”Generalized denoising auto-encoders as generative models.” Advances in Neural Information Processing Systems. 2013.

[7] Bengio, Yoshua, et al. ”Generalized denoising auto-encoders as generative models.” Advances in Neural Information Processing Systems. 2013.

[8] Bengio, Yoshua, Nicholas Leonard, and Aaron Courville. ”Estimating or propagating gradients through stochastic neurons for conditional computation.” arXiv preprint arXiv:1308.3432 (2013).