

# グラフスペクトルを用いた動的グラフに対する Graph Neural Network

大窪 諒<sup>1,a)</sup> 猪口 明博<sup>1</sup>

**概要:** 世の中にはグラフで表現できる対象物が多く存在し、その構造が時間とともに変化するものがある。これらの分析に Graph Neural Network (GNN) がよく用いられる。また、GNN の学習において、グラフスペクトルを組み込むとグラフの粗密情報を積極的に用いることができる。しかし、グラフスペクトルはラプラシアン行列の固有ベクトルであり、固有ベクトルはその逆方向も固有ベクトルであるため、その扱いには注意が必要である。本研究では、系列内の個々のグラフからグラフスペクトルを導出し、その方向性の時間変化を GNN の中で統一的に扱う方法を提案する。いくつかのデータセットを用いて本手法を評価した結果として、既存手法よりも予測精度が向上することを示す。

## 1. はじめに

世の中にはグラフ構造で表現できる対象物が多く存在している。例えば、ソーシャルネットワーキングサービスの人間関係は、アカウントをノード、アカウント同士の関係をエッジ、アカウントの持つ情報の特徴としたグラフとして表現できる。このグラフを分析すると将来関係を持ちそうなアカウント対を予測し、アカウントの推薦に役立てることができる。また、道路の状況も任意の地点をノード、それらの地点を結ぶ道路をエッジ、地点の交通量を特徴としたグラフで表現できる。このグラフを分析することで各地点の交通量を予測し、渋滞などを予測することができる。その他、ハイパーリンク構造や化合物、EC サイトもグラフで表現することができる。このようにグラフ構造で表現できるデータが多く存在し、これらのデータを分析すると有益な情報を得られる。グラフ構造を持ったデータの多くは時間経過で変化する動的なグラフである。この動的グラフとは、各時刻でのグラフの系列である。近年、このような動的グラフを対象とした分析手法が注目されている [10, 14, 15]。

近年の深層学習手法の発展によって、グラフデータの分析手法として Graph Neural Network (GNN) が注目されている [2, 13]。GNN はあるノード  $v$  の隣接ノードが持つ表現を畳み込み、新たな表現  $h_v$  を得る。GNN を用いた分析では、得られた表現  $h_v$  をもとに従来の機械学習法を適用

することでノード分類問題やエッジ予測問題を解く。GNN の代表例として Graph Attention Network (GAT) [13] が挙げられる。GAT はグラフ構造が変化しない静的なグラフを対象とした GNN である。GAT はノード  $v$  の畳み込みの際に Attention 機構を用い、ノード  $v$  の持つ表現  $h_v$  と類似した表現を持つ隣接ノードの表現を多く畳み込む。

GNN を動的グラフに適用する研究も進んでいる。例えば、Graph Attention Evolving Network (GAEN) [11] は Attention 係数を Gated Recurrent Unit (GRU) で計算することで、グラフの構造変化の時系列情報を GNN の学習に取り込む。それにより、GAEN は高い表現力を持つ表現を学習できる。

また、グラフのノードをクラスタリングする手法の一つにスペクトラルクラスタリング [5] がある。スペクトラルクラスタリングでは、グラフのラプラシアン行列から固有ベクトルを得る。この固有ベクトルをグラフフーリエ基底と呼ぶ。グラフフーリエ基底から各ノードに疎密情報を持った一つのベクトルを与え、そのベクトル形式のデータに  $k$ -means を適用し、クラスタを得る。

GNN では学習時に各ノードの隣接ノードの表現のみを考慮している。グラフフーリエ基底によって GNN の学習にグラフの疎密情報を用いることができれば、各ノードに隣接ノードの表現だけではなく、疎密情報から見た重要なノードの情報を考慮することができる。実際に、GNN にグラフフーリエ基底を活用する研究が進んでいる [1, 4, 8, 9]。しかし、グラフフーリエ基底の成分はラプラシアン行列の固有ベクトルであり、固有ベクトルはその逆方向のベクトルも固有ベクトルである。そのため、動的グラフの各時刻

<sup>1</sup> 関西学院大学大学院 理工学研究科  
Graduate School of Science and Technology, Kwansai Gakuin University

<sup>a)</sup> gqn04465@kwansai.ac.jp

のグラフに対して、独立してグラフフーリエ基底を計算すると、各時刻間でグラフフーリエ基底の向きが変わってしまうことがある。

本研究では、系列内の個々のグラフからグラフフーリエ基底を導出し、その方向性の時間変化を GNN の中で统一的に扱う方法を提案する。具体的には、前時刻のグラフから得られたラプラシアン行列の固有ベクトルと現在時刻のグラフから得られたラプラシアン行列の固有ベクトルの内積を用いて、前時刻と現在時刻との固有ベクトルとの向きの違いを判断し、ベクトルの向きが異なっていた場合に修正する。本手法によってグラフフーリエ基底の向きをそろえ、動的グラフを対象とした既存の GNN に組み込むことでエッジ予測問題において予測精度の向上を目指す。また、いくつかのデータセットを用いて本手法を評価した結果として既存手法よりも予測精度が向上することを示す。

## 2. 問題設定

本研究では、ノードが特徴を持つ時刻  $t$  のグラフを  $G^t = (V^t, E^t, X^t)$  で表す。ここで、 $V^t = \{v_1^t, v_2^t, \dots, v_n^t\}$  は時刻  $t$  のノード集合、 $E^t \subseteq V^t \times V^t$  は時刻  $t$  のエッジ集合、 $X = \{\mathbf{x}_1^t, \mathbf{x}_2^t, \dots, \mathbf{x}_n^t\}$  は時刻  $t$  にノードが持つ特徴の集合である。ノード  $v_i$  自身を含む隣接ノード集合を  $N_i = \{v_j \mid (v_i, v_j) \in E\} \cup \{v_i\}$  とする。ここで、ある時刻において存在していないノードはどのノードともエッジを持たない孤立点として扱い、任意の  $t$  に対して  $|V^t| = n$  となる。また、グラフ系列の連続する 2 つのグラフ  $G^t$  と  $G^{t+1}$  の構造の変化は大きくないと仮定する。本研究で扱う問題は時刻  $T$  までのグラフ集合  $\{G^1, G^2, \dots, G^T\}$  が与えられたとき、次の時刻  $T+1$  における一部のノード対  $\Phi \subseteq V^{T+1} \times V^{T+1}$  に対して、エッジの有無を予測することである。

## 3. 関連研究

### 3.1 Graph Neural Network

GNN の代表的な手法として GAT を紹介する。GAT は各ノード  $v_i$  の隣接ノードの表現を不均一に畳み込む。GAT の畳み込みは以下の式で表される。

$$\mathbf{a}_i^l = \sum_{v_j \in N_i} \alpha_{ij} \mathbf{h}_j^{l-1} \quad (1)$$

$$\mathbf{h}_i^l = \sigma(W^l \mathbf{a}_i^l) \quad (2)$$

ここで、 $\alpha_{ij}$  はノード  $v_i$  と  $v_j$  間の Attention 係数、 $\mathbf{h}_i^l$  は  $l$  層目におけるノード  $v_i$  が持つ表現、 $W^l$  は学習パラメータ、 $\sigma$  は活性化関数である。式 (1) では  $N_i$  内のノードが持つ表現を集約しており、式 (2) では集約した表現にニューラルネットワークを適用して各ノードの新たな表現を得る。GAT は  $\mathbf{h}_i^0 = \mathbf{x}_i$  として、この畳み込みを  $L$  回繰り返して、表現  $\mathbf{h}_i^L$  をノード  $v_i$  の表現として出力する。GAT は隣接

ノードの表現の中で、Attention 係数が高いノードの表現を多く畳み込む。ここで、Attention 係数は

$$\hat{\alpha}_{ij}^l = \sigma(\mathbf{a} \cdot [W^l \mathbf{h}_i^l \parallel W^l \mathbf{h}_j^l]) \quad (3)$$

$$\alpha_{ij}^l = \frac{\exp(\hat{\alpha}_{ij}^l)}{\sum_{v_k \in N_i} \exp(\hat{\alpha}_{ik}^l)} \quad (4)$$

で計算される。 $\alpha_{ij}^l$  は自身の持つ表現と隣接ノードの表現の重要度を表す。また、 $\parallel$  はベクトルの連結操作を表す。式 (4) は Softmax 関数であり、式 (3) で計算した重要度の合計が 1 になるように正規化する。この Attention 係数によって畳み込みの際にノードの表現を不均一に集約する。

次に、動的グラフを対象とした GNN である GAEN を紹介する。GAEN は 2 つのアイデアで GAT を動的グラフに対して拡張している。一つ目のアイデアは Attention 係数を計算する際に時系列情報を取り込むことである。GAEN は Attention 係数を計算する際に使う重みを GRU で学習させることで Attention 係数に時系列情報を取り込んでいる。GAEN の Attention 係数は以下で計算される。

$$W^t = GRU(A^t, W^{t-1})$$

$$\hat{\alpha}_{ij} = \sigma(\mathbf{a} \cdot [W^t \mathbf{h}_i \parallel W^t \mathbf{h}_j]) \quad (5)$$

$$\alpha_{ij} = \frac{\exp(\hat{\alpha}_{ij})}{\sum_{v_k \in N_i} \exp(\hat{\alpha}_{ik})}$$

GRU に現在の状態である隣接行列と過去の状態である前時刻の重み  $W^{t-1}$  を入力し、現在時刻の重みを出力している。GRU は、その内部に学習可能なパラメータを持ち、これによって Attention 機構の重みが過去の状態を考慮した重みになるため、Attention 係数の計算に時系列情報を取り込むことができる。二つ目のアイデアは畳み込みの際に時系列情報を取り込むことである。まず、時刻  $t$  までのグラフの隣接行列を新しい軸方向に積み重ねたテンソルを作成する。それをテンソル分解し、時刻  $t$  までのノード変化を表した行列  $C^t$  を得る。GAEN の畳み込みは

$$\mathbf{a}_i^t = \sum_{v_j \in N_i} \alpha_{ij} C_{ij}^t \mathbf{x}_j^t \quad (6)$$

$$\mathbf{h}_i = \sigma(W^t \mathbf{a}_i^t) \quad (7)$$

で計算される。ここで、 $C_{ij}^t$  は行列  $C^t$  の  $(i, j)$  要素を表す。 $C_{ij}^t$  はノード  $v_i$  と  $v_j$  間のノードの変化パターンの類似度を表すため、 $C^t$  を畳み込みに組み込むことによって GAEN は畳み込みの際にノードの変化を考慮できる。この二つのアイデアで GAT を動的グラフにも適用できるように拡張している。

### 3.2 スペクトラルクラスタリング

スペクトルクラスタリングはグラフのノードクラスタリング問題を解く手法である。スペクトルクラスタリングは各ノードに疎密情報を持った  $k$  次元のベクトルを与え、

このベクトルに  $k$ -means を適用することでノードクラスタリングをする。具体的には、グラフの隣接行列と次数行列をそれぞれ  $A$  と  $D$  とする。また、 $D - A$  をラプラシアン行列  $L$  とし、 $L$  の  $k$  番目に小さい固有値に対応する固有ベクトルを  $\mathbf{u}_k$  とする。この  $\mathbf{u}_k$  を  $k$  番目のグラフフーリエ基底と呼ぶ。ノード  $v_i$  に与えるベクトルは、 $k$  個のグラフフーリエ基底の第  $i$  成分からなる  $k$  次のベクトルである。この手順で得られたベクトルデータに対して  $k$ -means を適用することでノードクラスタリングする。

#### 4. 提案手法

本稿では、グラフフーリエ基底がもつグラフの疎密情報を、グラフ系列を学習する GNN に、取り込むことを考える。文献 [5] では、

- (a) スペクトラルクラスタリングはグラフを適度な大きさの部分グラフに分割することができる
- (b) 2つの行列がわずかに違う場合、それらの行列から得られる固有ベクトルの違いは小さい

ことが述べられており、我々はこれらの性質を活用する。

(a) より、グラフフーリエ基底の低周波成分はグラフの疎密情報を持つと言える。また、(b) より、動的なグラフにおいて、連続する時刻間でその構造の変化がわずかな場合、連続する 2 時刻のグラフに対するラプラシアン行列の変化も小さく、グラフフーリエ基底の変化も小さい。そのため、わずかな変化をするグラフに対してはグラフフーリエ基底で時間的な変化を捉えられると考えられる。この時間的な変化を動的グラフを対象とした GNN で捉えることができればより高い表現力を獲得できると考える。

##### 4.1 グラフフーリエ基底を GNN に組み込む際の課題

系列内の個々のグラフに対してグラフフーリエ基底を計算する場合、その取り扱いに注意が必要である。図 1 は動的に変化するグラフと、それらのグラフフーリエ基底の例である。この図の左側は時刻  $t$  と時刻  $t+1$  のグラフを示している。右側ではそれらのグラフのラプラシアン行列の固有ベクトルをそれぞれ計算し、各ノードに 2 次元の疎密情報を持ったベクトルを与え、図示した。この図では連続する時刻間でのグラフの変化が小さいのにもかかわらず、与えられたグラフフーリエ基底は大きく異なる。これはラプラシアン行列の固有ベクトル  $\mathbf{u}_k$  の逆方向  $-\mathbf{u}_k$  も固有ベクトルであるからである。このように、連続する 2 時刻のグラフにおいてグラフフーリエ基底が大きく変化してしまうことがある。

##### 4.2 グラフフーリエ基底の向きの一手法

本研究では動的グラフを扱う際にグラフフーリエ基底を GNN に組み込む手法を提案する。動的グラフの各系列に対してグラフフーリエ基底を計算すると前時刻と現在時刻

##### Algorithm 1: グラフフーリエ基底の向きの一手法

---

Inputs: Graph fourier bases  $\{\bar{S}^1, \bar{S}^2, \dots, \bar{S}^T\}$   
Outputs: Unified graph fourier bases  $\{S^1, S^2, \dots, S^T\}$

```

1: for  $t \in [1, T]$  do
2:    $\{\mathbf{u}_1^t, \mathbf{u}_2^t, \dots, \mathbf{u}_k^t\} \leftarrow \mathbf{u}_k^t$  is the  $k$ -th row of  $\bar{S}^t$ 
3:   if  $t > 1$  then
4:     for  $i \in [1, k]$  do
5:       if  $\mathbf{u}_i^t \cdot \mathbf{u}_i^{t-1} < 0$  then
6:          $\mathbf{u}_i^t \leftarrow -\mathbf{u}_i^t$ 
7:       end if
8:     end for
9:   end if
10:   $S^t \leftarrow$  matrix of the vectors  $\mathbf{u}_1^t, \mathbf{u}_2^t, \dots, \mathbf{u}_k^t$  as row
11: end for
12: return  $S = \{S^1, S^2, \dots, S^T\}$ 

```

---

でグラフフーリエ基底の向きが異なる可能性がある。そこで、前時刻で計算したラプラシアン行列の固有ベクトルと現在時刻で計算したラプラシアン行列の固有ベクトルとの内積を用いて、各時刻でグラフフーリエ基底の軸の向きが反転している可能性がないかを確認し、反転していた場合は軸の向きを統一させる。具体的には

$$f(\mathbf{u}_i^t, \mathbf{u}_i^{t-1}) = \text{sign}(\mathbf{u}_i^t \cdot \mathbf{u}_i^{t-1}) \quad (8)$$

をグラフフーリエ基底にかける。ここで、 $\mathbf{u}_i^t$  は時刻  $t$  の  $i$  番目のグラフフーリエ基底を表す。 $f(\mathbf{u}_i^t, \mathbf{u}_i^{t-1})$  は前時刻のグラフフーリエ基底と現在時刻のグラフフーリエ基底の内積を計算し、負の値を取るときに  $-1$  を、正の値を取るときに  $1$  を返す。その後、 $f(\mathbf{u}_i^t, \mathbf{u}_i^{t-1}) \times \mathbf{u}_i^t$  を新たに現在時刻のグラフフーリエ基底  $\mathbf{u}_i^t$  とする。前時刻と現在時刻のグラフフーリエ基底の内積が負であった場合に  $-1$  をかけることでグラフフーリエ基底の向きを反転させる。

##### 4.3 提案手法の擬似コード

本小節では前時刻のグラフフーリエ基底と現在時刻のグラフフーリエ基底の向きを確認し、異なっていた場合に修正する手法を Algorithm 1 に示す。Algorithm 1 の入力のは向きが統一化されていないグラフフーリエ基底を行方向に結合した行列の系列  $\{\bar{S}^1, \bar{S}^2, \dots, \bar{S}^T \mid \bar{S}^t \in \mathbb{R}^{k \times n}\}$ 、出力は向きが統一化がされているグラフフーリエ基底を行方向に結合した行列の系列  $\{S^1, S^2, \dots, S^T \mid S^t \in \mathbb{R}^{k \times n}\}$  である。Algorithm 1 の 2 行目では時刻  $t$  のグラフフーリエ基底を取り出している。5 行目では前時刻と現在時刻の  $i$  番目のグラフフーリエ基底同士の内積を計算する。その内積が負である場合、6 行目で現在時刻の  $i$  番目のグラフフーリエ基底に  $-1$  をかけて向きを反転させる。10 行目は時刻  $t$  で向きを統一したグラフフーリエ基底を行方向に結合した行列を作成する。各時刻のグラフから得られるグラフフーリエ基底にこの操作を適用し、時刻間でのグラフフーリエ基底の向きを統一させる。この手順で得られた向きの統一がされたグラフフーリエ基底を GNN に組み込む。

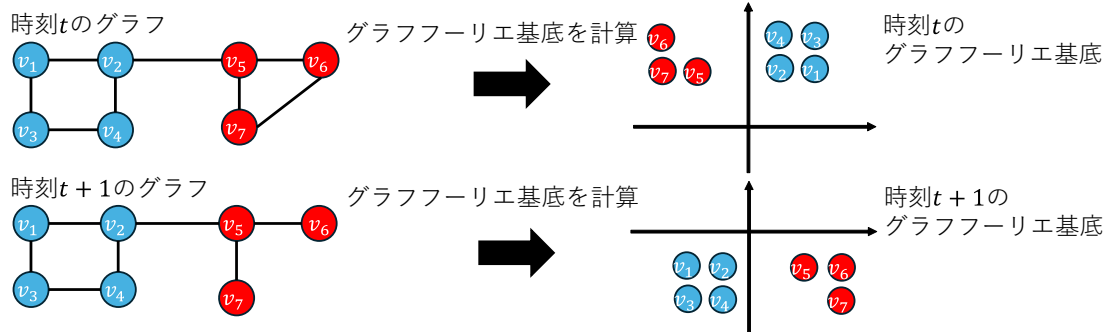


図 1: 動的グラフに対するグラフフーリエ基底を計算する際に固有ベクトルが反転する影響の例

#### 4.4 本手法の GNN への組み込み

本小節では、向きを統一化したグラフフーリエ基底の GNN への組み込み方について議論する。グラフフーリエ基底はグラフの疎密情報を持っている。そのため、グラフフーリエ基底を GNN に組み込むと隣接ノードの表現だけでなく、疎密情報からノード同士が密につながっているノードの情報を得ることができる。このことから GNN にグラフフーリエ基底を組み込むことは GNN の表現力の向上に役立つと考えられる。

本論文ではグラフフーリエ基底を組み込む GNN として GAEN を試す。GAEN は Attention 係数を計算する際の重みを GRU で学習する。本論文では以下に示すように GRU の入力の現在の状態を本手法で向きを統一化したグラフフーリエ基底と隣接行列を結合した行列とし、過去の状態を前時刻の重みとする。

$$\begin{aligned}
 W^t &= GRU([S^t || A^t], W^{t-1}) \\
 \hat{\alpha}_{ij} &= \sigma(\mathbf{a} \cdot [W^t \mathbf{h}_i || W^t \mathbf{h}_j]) \\
 \alpha_{ij} &= \frac{\exp(\hat{\alpha}_{ij})}{\sum_{v_k \in N_i} \exp(\hat{\alpha}_{ik})}
 \end{aligned} \quad (9)$$

GRU に隣接行列だけでなく、グラフフーリエ基底も入力することで、Attention 係数にグラフの疎密情報を加味させる。これにより、GAEN の畳み込みで隣接ノードの特徴からだけでなく、疎密情報から見た際の似たノードの情報を考慮した畳み込みができる。

## 5. 評価実験

### 5.1 実験設定

本節では提案手法を取り入れた GNN の精度がどの程度上昇するのかをベンチマークデータセットを用いた実験で示す。表 1 は本論文で用いたベンチマークデータセットの概要である。Enron [11] は、Enron 社の中核となる従業員の電子メールのやり取りを表現したデータセットであり、従業員をノード、メールのやり取りをエッジとしたグラフとする。School [12] は、232 人の子供と 10 人の教師間の直接の対話を表現したデータセットであり、教師や子供をノード、対話をエッジとしたグラフとする。UCI [6] は、オ

表 1: ベンチマークデータセットの概要

データセット名	ノード数 $n$	エッジ数 $\sum_{t=1}^{T+1}  E^t $	時刻数 $T+1$
Enron	143	2,347	10
School	242	20,019	40
UCI	1,890	16,822	13

ンライン上のソーシャルネットワークでのやり取りを表現したデータセットであり、ユーザーをノード、ユーザー間でのやり取りをエッジとしたグラフとする。これらのデータセットで GNN の性能を評価する。

本稿では動的グラフを対象とした既存手法として GAEN と EvolveGCN [7] を用いた。EvolveGCN は GAEN と同様に、動的グラフを対象とした GNN で、GRU を用いて Graph Convolutional Network (GCN) [2] の畳み込みに使われる重みを系列データとして学習することで、時系列情報を GCN の畳み込みに取り込んでいる。また、提案手法の様々なバリエーションを考え、手法<sub>NoAdj</sub> や手法<sub>Basic</sub> とも比較する。それらの違いは、以下に示す  $W^t$  の導出における GRU の入力の違いによるものである。

- 提案手法:  $W^t = GRU([S^t || A^t], W^{t-1})$
- 手法<sub>NoAdj</sub>:  $W^t = GRU(S^t, W^{t-1})$
- 手法<sub>Basic</sub>:  $W^t = GRU([\bar{S}^t || A^t], W^{t-1})$
- GAEN:  $W^t = GRU(A^t, W^{t-1})$

手法<sub>NoAdj</sub> は式 (9) における隣接行列の必要性を検証するためのものである。また、手法<sub>Basic</sub> はグラフフーリエ基底の向きを統一したことによる効果を検証するためのものであり、グラフフーリエ基底の向きを統一していない行列を用いる。

実験では各データセットに対してエッジ予測問題を解き、Area Under Curve (AUC) を使って予測性能を評価する。時刻  $T$  までのグラフから学習した表現に対して時刻  $T+1$  におけるエッジを予測するために GNN の最終層で以下に示すロジスティック回帰で学習する。GNN で得られたノード  $v_i$  と  $v_j$  のそれぞれの表現  $\mathbf{h}_i$  と  $\mathbf{h}_j$  に対して、それらのノード間のエッジの予測確率  $p(e_{ij})$  は

$$p(e_{ij}) = \sigma_s(\mathbf{w} \cdot [\mathbf{h}_i \odot \mathbf{h}_j]) \quad (10)$$

で計算した。ここで、 $\sigma_s$  はシグモイド関数、 $w$  は学習パラメータ、 $\odot$  はアダマール積である。 $p(e_{ij})$  が 0.5 以上であればエッジが存在すると予測する。学習器のハイパーパラメータを調整するためにエッジ  $E^{T+1}$  のうち 20% を検証用データとして使い、残りのエッジを 25% と 75% に分割して学習とテストを行う。

## 5.2 実験結果

表 2 は各データセットに対してエッジ予測問題を解いた際の AUC を表している。太文字は各データセットでの最高精度である。また、括弧の中の数字は用いたグラフフーリエ基底の個数  $k$  である。全てのデータセットにおいて提案手法の AUC が EvolveGCN や GAEN の AUC よりも高くなった。これは GNN の学習にグラフフーリエ基底を組み込むことへの有用性を示している。また、全てのデータセットにおいて手法  $NoAdj$  の AUC よりも提案手法の AUC のほうが高い値になった。これは、グラフフーリエ基底は隣接行列から抽出されたものではあるが、隣接行列とグラフフーリエ基底がもつ情報が必ずしも同じではないことを示唆している。また、Enron と UCI において手法  $Basic$  の AUC よりも提案手法と手法  $NoAdj$  の AUC のほうが高い結果となった。これはグラフフーリエ基底の向きを統一することでグラフの構造の変化を GNN が捉えやすくなり、GNN の学習において良い影響を及ぼしたためだと考える。

次に、各データセットのエッジ  $\Phi \cap E^{T+1}$  に対して、各手法がエッジがあると予測した割合、つまり再現率を確認した。表 3 はその再現率を示したものである。EvolveGCN は AUC は低いですが、その再現率は二つのデータセットで最も高い値になった。これは EvolveGCN が積極的にエッジがあると予測したためである。一方で、エッジがないノード対に対してもエッジがあると予測することで、適合率は

表 2: 各ベンチマークデータセットでのエッジ予測の AUC 値 (括弧内の数値は、提案手法における  $k$  の値)

	Enron	School	UCI
EvolveGCN	0.831	0.932	0.828
GAEN	0.912	0.958	0.974
提案手法	<b>0.978</b> (100)	<b>1.000</b> (175)	<b>0.994</b> (50)
手法 $NoAdj$	0.936 (50)	0.971 (125)	0.984 (125)
手法 $Basic$	0.914 (50)	<b>1.000</b> (50)	0.954 (25)

表 3: 各ベンチマークデータセットでの再現率 (括弧内の数値は、提案手法における  $k$  の値)

	Enron	School	UCI
EvolveGCN	<b>0.964</b>	0.942	<b>0.978</b>
GAEN	0.831	0.977	0.886
提案手法	0.913 (100)	<b>1.000</b> (175)	0.950 (50)
手法 $NoAdj$	0.823 (50)	0.977 (125)	0.924 (125)
手法 $Basic$	0.882 (50)	<b>1.000</b> (50)	0.961 (25)

必ずしも高くなく、AUC が低い値になった。全てのデータセットにおいて GAEN の再現率よりも提案手法の再現率のほうが高くなった。これは、グラフフーリエ基底を組み込むことで、実際にエッジがあるノード対に対して積極的にエッジがあると予測するようになったことを示している。また、UCI を除いて手法  $NoAdj$  の再現率は GAEN の再現率と同程度であった。これはグラフフーリエ基底の低周波成分が隣接行列の情報を捉えたためだと考える。

次に、提案手法と GAEN の  $(v_i, v_j) \in \Phi$  に対する  $p(e_{ij})$  の分布を確認した。図 2 は各データセットに対する提案手法と GAEN の  $p(e_{ij})$  の分布を箱ひげ図で示したものである。positive edge と書かれた各図の左の箱ひげ図は  $(v_i, v_j) \in \Phi \cap E^{T+1}$  に対する  $p(e_{ij})$  の分布を、negative edge 側の箱ひげ図は  $(v_i, v_j) \in \Phi \setminus E^{T+1}$  に対する  $p(e_{ij})$  の分布を表している。School と UCI に対して、提案手法は GAEN よりもエッジがあるノード対とエッジがないノード対に対する予測確率の分布の分散が小さい。これは、提案手法が GAEN よりもエッジがあるノード対とエッジがないノード対を分ける能力が高いことを示している。また、提案手法は予測確率の分布の分散を小さくするか、エッジのあるノード対に対する予測確率の上限を上げ、エッジのないノード対に対する予測確率の下限を下げることで、予測確率の分布の重なりを小さくしている。この結果から、提案手法は、エッジのあるノード対は表現空間上で一定の距離を保つ、もしくはより近いベクトルを割り当てる効果があると考えられる。さらに、提案手法はエッジのないノード対に対しては表現空間上でエッジのあるノード対よりも遠いベクトルを割り当てる効果があると考えられる。

## 6. まとめ

GNN にグラフの疎密情報を組み込むことで、GNN の学習に隣接ノードだけでなく、疎密情報を考慮することができる。しかし、グラフフーリエ基底はグラフのラプラシアン行列の固有ベクトルであり、固有ベクトルの逆もまた固有ベクトルとなるため、連続する時刻間でグラフフーリエ基底の向きが変わってしまう可能性がある。そこで、本論文では連続する時刻間でのグラフフーリエ基底の向きを統一し、GNN に組み込む手法を提案した。具体的には、前時刻と現在時刻のグラフフーリエ基底同士の内積を計算し、それが負の値であるときに向きを反転させた。評価実験より、グラフフーリエ基底を組み込んだほうがエッジの予測精度に優れているということが分かった。本論文ではグラフフーリエ基底を組み込む GNN として GAEN を試したが、本手法によって得られた統一化されたグラフフーリエ基底は、動的グラフを解析する他の GNN にも適用可能である。今後の課題として、他の GNN に本手法を組み込むことを検討している。

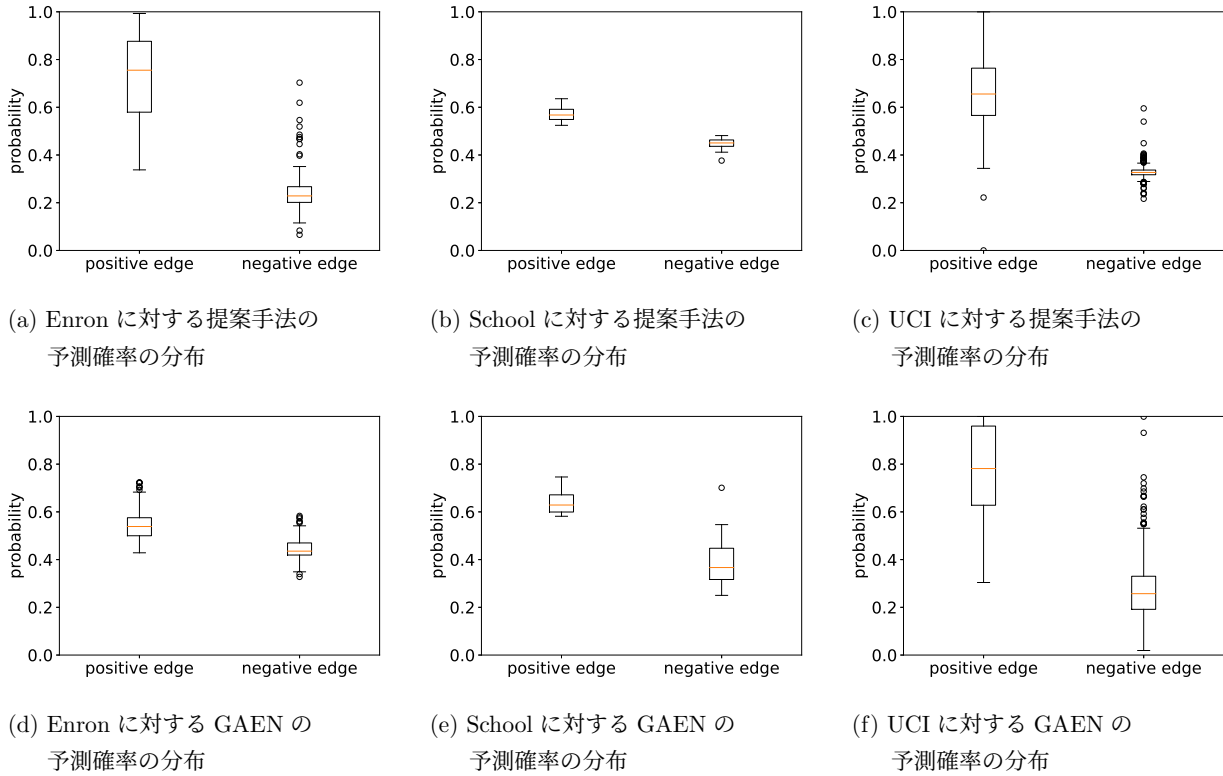


図 2: 各データセットに対する各手法の予測確率の分布

## 参考文献

- [1] Vijay Prakash Dwivedi and Xavier Bresson. “A Generalization of Transformer Networks to Graphs.” *AAAI Workshop on Deep Learning on Graphs*, 2021.
- [2] Thomas Kipf and Max Welling. “Semi-Supervised Classification with Graph Convolutional Networks.” *Proc. of Int’l Conf. on Learning Representations*, 2017.
- [3] Bryan Klimt and Yiming Yang. “Introducing the Enron Corpus.” *Proc. of Int’l Conf. on Email and Anti-Spam*, 2004.
- [4] Devin Kreuzer, Dominique Beaini, William L. Hamilton, Vincent Létourneau, and Prudencio Tossou. “Rethinking Graph Transformers with Spectral Attention.” *Proc. of Advances in Neural Information Processing Systems*, 2021, pp. 21618–21629.
- [5] Ulrike von Luxburg. “A Tutorial on Spectral Clustering.” *Statistics and Computing*, 2007, pp. 395–416.
- [6] Pietro Panzarasa, Toe Opsahl, and Kathleen M. Carley. “Patterns and Dynamics of Users’ Behavior and Interaction: Network Analysis of an Online Community.” *Journal of the American Society for Information Science and Technology*, 2009, vol. 60, pp. 911–932.
- [7] Aldo Pareja, Giacomo Domeniconi, Jie Chen, Tengfei Ma, Toyotaro Suzumura, Hiroki Kanezashi, Tim Kaler, Tao B. Schardl, and Charles E. Leiserson. “EvolveGCN: Evolving Graph Convolutional Networks for Dynamic Graphs.” *Proc. of the AAAI Conf. on Artificial Intelligence*, 2020, pp. 5363–5370.
- [8] Jiezhong Qiu, Qibin Chen, Yuxiao Dong, Jing Zhang, Hongxia Yang, Ming Ding, Kuansan Wang, and Jie Tang. “GCC: Graph Contrastive Coding for Graph Neural Network Pre-Training.” *Proc. of Conf. on Knowledge Discovery and Data Mining*, 2020, pp. 1150–1160.
- [9] Ladislav Rampásek, Michael Galkin, Vijay Prakash Dwivedi, Anh Tuan Luu, Guy Wolf, and Dominique Beaini. Recipe for a General, Powerful, Scalable Graph Transformer. *Proc. of Advances in Neural Information Processing Systems*, 2022.
- [10] Aravind Sankar, Yanhong Wu, Liang Gou, Wei Zhang, and Hao Yang. “DySAT: Deep Neural Representation Learning on Dynamic Graphs via Self-Attention Networks.” *Proc. of Int’l Conf. on Web Search and Data Mining*, 2020, pp. 519–527.
- [11] Min Shi, Yu Huang, Xingquan Zhu, Yufei Tang, Yuan Zhuang, and Jianxun Liu. “GAEN: Graph Attention Evolving Networks.” *Proc. of Int’l Joint Conf. on Artificial Intelligence*, 2021, pp. 1541–1547.
- [12] Juliette Stehlé, Nicolas Voirin, Alain Barrat, Ciro Cattuto, Lorenzo Isella, Jean-François Pinton, Marco Quaggiotto, Wouter van den Broeck, Corinne Régis, Bruno Lina, and Philippe Vanhems. “High-Resolution Measurements of Face-to-Face Contact Patterns in a Primary School.” *PloS one*, 2011, vol. 6, pp. e23176.
- [13] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. “Graph Attention Networks.” *Proc. of Int’l Conf. on Learning Representations*, 2018.
- [14] Leshanshui Yang, Clément Chatelain, and Sébastien Adam. “Dynamic Graph Representation Learning with Neural Networks: A Survey.” *IEEE Access*, 2024, vol. 12, pp. 43460–43484.
- [15] Li Zheng, Zhenpeng Li, Jian Li, Zhao Li, and Jun Gao. “AddGraph: Anomaly Detection in Dynamic Graph Using Attention-based Temporal GCN.” *Proc. of Int’l Joint Conf. on Artificial Intelligence*, 2019, pp. 4419–4425.