

# TTSP 項グラフパターンに対する学習済み高精度 GCN を オラクルとする質問学習モデルにおける質問回数削減効果

岡本悠生<sup>1,a)</sup> 内田智之<sup>1,b)</sup> 正代隆義<sup>2,c)</sup> 松本哲志<sup>3,d)</sup> 鈴木祐介<sup>1,e)</sup> 宮原哲浩<sup>1,f)</sup>

**概要:** 質問学習は、学習者が教師（オラクル）に質問を行い、その応答から概念を同定する手法である。本研究では、電気ネットワークなどのデータモデルとして用いられる TTSP (Two-Terminal Series Parallel) グラフの集合を定義する形式体系  $pFS_{TTSP}$  を概念の可視化表現として用いる。さらに、この集合を学習した高精度 GCN をオラクルとし、GCN が獲得した概念を  $pFS_{TTSP}$  上で可視化する質問学習アルゴリズムを提案する。人工データによる評価実験では、質問回数削減手法を組み込むことで可視化精度が向上することを確認したため、その結果を報告する。

**キーワード:** 機械学習・データマイニング, 知識発見, 計算論的学習理論

## Effectiveness of query reduction in query learning using a high-accuracy GCN trained on TTSP-term graph patterns as an oracle

**Abstract:** Query learning identifies concepts by submitting queries to an oracle and interpreting its responses. This study employs  $pFS_{TTSP}$ , a formal system defining TTSP graphs used in electrical networks, as a representation for visualizing learned concepts. A high-accuracy GCN trained on this graph set serves as the oracle, and we propose a query learning algorithm that visualizes the GCN's acquired concepts within the  $pFS_{TTSP}$  framework. Experiments on synthetic data show that incorporating a query-reduction method improves visualization accuracy, and we report these findings.

**Keywords:** Machine Learning, Data Mining, Knowledge Discovery, Computational Learning Theory

### 1. はじめに

機械学習手法の1つである深層学習は、人間の神経細胞を模したニューラルネットワークを活用することで、大量のデータに内在する特徴を学習することができる。それによりデータの分類、予測を高精度で行うことができ、画像処理や自然言語処理など多くの分野で活用されている。そ

の一方で、なぜその結果を出したかという予測根拠については透明性・説明性が低いブラックボックス型の学習モデルであることが知られている。特に医療分野や自動運転など予測根拠が重要な分野での活用が難しいのが現状である。近年、ブラックボックス型の学習モデルの予測根拠を可視化する研究が活発に行われている。特に、人間関係のネットワークや分子構造などのグラフでモデル化できるデータ（グラフ構造データという）を扱うことができる深層学習モデルであるグラフニューラルネットワーク（Graph Neural Networks, GNNs）の研究が盛んに行われている。ラベルづけされたグラフの集合を学習させた GNN に、ラベルづけされていないグラフのラベルを予測させることで、グラフ分類を行うことができる。このため、グラフ構造データからの構造的な特徴抽出に GNN を活用しようとする研究が活発化している。

Angluin[1] により提唱された質問学習モデルは、常に正

<sup>1</sup> 広島市立大学  
hiroshima City University  
<sup>2</sup> 福岡工業大学  
Fukuoka Institute of Technology  
<sup>3</sup> 東海大学  
Tokai University  
a) mj67002@e.hiroshima-cu.ac.jp  
b) uchida@hiroshima-cu.ac.jp  
c) shodai@fit.ac.jp  
d) matsumoto@tokai.ac.jp  
e) y-suzuki@hiroshima-cu.ac.jp  
f) miyares25@hiroshima-cu.ac.jp

答を返す教師(オラクル)に学習者が質問を繰り返すことで、教師の有する概念を同定する学習手法である。内田ら [2] は、順序項木パターンを定義するプリミティブ形式体系を用いた、順序木言語を同定する質問学習アルゴリズムを提案した。また質問学習モデルは、巨大なデータベースに対する質問を用いたデータマイニング戦略の数学的モデルである。未知なるデータがデータベース上に存在するかどうか問い合わせる所属性質問はデータベースを用いたデータマイニングでよく用いられる。ゆえに、巨大なデータベースから特徴を抽出するためには、より少ない所属性質問を用いた、特徴を同定するデータマイニングアルゴリズムが求められる。頂点の更新の際に隣接する頂点から特徴を取り込む「畳み込み演算」を行うことで頂点の特徴量に隣接関係を反映させられる GNN であるグラフ畳み込みネットワーク (Graph Convolutional Networks, GCNs)[3] がある。石灘ら [4] は、グラフ構造データの一つである無順序木構造データを学習させた高精度 GCN をオラクルとし質問学習モデルの解析を行っている。さらに、東山ら [5] は、順序木構造データを対象とした学習済み高精度 GCN に対する質問学習アルゴリズムを提案し、分類精度を向上させる手法を提案している。TTSP グラフ (Two-Terminal Series Parallel graph) は、電気ネットワークやスケジューリングをコンピュータで扱う際にデータモデルとしてよく用いられる。TTSP グラフは、並列操作と直列操作と呼ばれる操作を再帰的に適用することで得られる多重辺を許したサイクルを持たない連結有向グラフである。本論文では、TTSP グラフでモデル化できるグラフ構造データを対象とした GCN をオラクルとする質問学習アルゴリズムを提案し、人工データを用いた評価実験を行ってその有用性について示すことを目的とする。この目的を達成するために、まず TTSP グラフの集合を定義するグラフ文法としてプリミティブ形式 TTSP グラフ体系 (Primitive Formal TTSP Graph System,  $pFS_{TTSP}$ ) を提案する。 $pFS_{TTSP}$  は、変数と呼ばれる特別な辺を持つ TTSP グラフである TTSP 項グラフパターンを項としてもつ論理プログラミング体系である形式グラフ体系 (Formal Graph System, FGS) [6] である。 $pFS_{TTSP}$  プログラムとその言語の例を図 1 に与える。図 1 の  $rule3$  は直列操作に相当するルールを表し、 $rule4$  は並列操作に相当するルールを表す。論理プログラミングと同様に、 $pFS_{TTSP}$  プログラム  $\Gamma$  と述語記号  $r$  を用いて定義されるグラフ言語  $L(\Gamma, r)$  とは、 $\Gamma$  を用いてゴール ' $\leftarrow r(G)$ ' から始まり空ゴールで終わる導出が存在する TTSP グラフ  $G$  全体の集合のことをいう。この言語  $L(\Gamma, r)$  を  $\Gamma$  と  $r$  による  $pFS_{TTSP}$  言語という。TTSP グラフの全体集合を定義する  $pFS_{TTSP}$  言語の部分集合を学習させた高精度な GCN をオラクルとして用い、そのオラクルへの所属性質問回数の効率化を図った質問学習アルゴリズムを構築する。さらに、可視化精度に関して

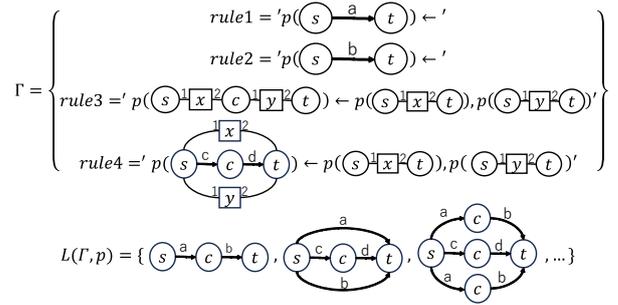


図 1:  $pFS_{TTSP}$  プログラム  $\Gamma$  と  $pFS_{TTSP}$  言語  $L(\Gamma, p)$

従来手法との比較評価実験を行い、質問回数削減による本手法の有効性を検証する。

## 2. プリミティブ TTSP グラフ体系とそのグラフ言語

並列操作と直列操作を帰納的に定義して得られる多重辺を許すサイクルを持たない連結有向グラフを **TTSP グラフ** (Two-Terminal Series Parallel Graph)[6] という。

$\Sigma$  を有限アルファベット,  $\Lambda$  と  $X$  を無限アルファベットとする。グラフ  $G$  に対して  $V(G)$  は  $G$  の頂点集合,  $E(G)$  は  $G$  の辺集合という。また任意の 2 頂点  $u_0, u_1 \in V(G)$  について、 $(u_0, u_1)$  は  $u_0, u_1$  間の辺を表す。また  $S$  を集合、もしくはリストとすると、 $|S|$  は  $S$  の要素数を表す。ここで、以下のように定義されるグラフパターンを **TTSP 項グラフパターン** という。

$G$  を TTSP グラフ,  $H_G$  を  $E(G)$  の部分集合とする。このとき、 $V = V(G), E = E(G) \setminus H_G, H = H_G$  を用いて表される 3 個組  $(V, E, H)$  を **TTSP 項グラフパターン** という。また TTSP 項グラフパターン  $g$  について、 $(u_0, u_1) \in H_G$  を  $g$  の変数と呼び、 $[u_0, u_1]$  と表記する。 $G$  の各頂点は  $\Sigma$  中の記号、辺は  $\Lambda$  中の記号でそれぞれラベル付けされる。また、変数  $h = [u_0, u_1]$  は記号  $x \in X$  でラベル付けされる。このとき、 $x$  は変数ラベル、 $u_0$  を親ポート、 $u_1$  を子ポートと呼ぶ。

TTSP 項グラフパターン  $g$  に対して、 $V(g)$  は  $g$  の頂点集合、 $E(g)$  は  $g$  の辺集合、 $H(g)$  は  $g$  の変数集合を表す。また、 $v \in V(g)$  の頂点ラベルは  $\varphi_g(v) \in \Sigma$ 、 $e \in E(g)$  の辺ラベルは  $\psi_g(v) \in \Lambda$ 、 $h \in H(g)$  の変数ラベルは  $\lambda_g(v) \in X$  を用いて表記する。各変数に対して、変数ラベルが重複しない TTSP 項グラフパターンを **線形 TTSP 項グラフパターン** という。本論文では線形 TTSP 項グラフパターンのみを扱うため、以降線形 TTSP 項グラフパターンを単に TTSP 項グラフパターン、変数を持たない TTSP 項グラフパターンを単に TTSP グラフということにする。また、TTSP 項グラフパターン全体を  $GP_{TTSP}$ 、TTSP グラフ全体を  $GT_{TTSP}$  と表現する。

$G$  を、少なくとも 2 つの頂点を持つ TTSP グラフ、 $g$  を

少なくとも1つの変数を持つ TTSP 項グラフパターン,  $x$  を  $g$  中の変数ラベル,  $u_0$  を  $G$  のソース,  $u_1$  を  $G$  のシンクとする. 2つの異なる  $G$  中の頂点のリストを  $\sigma = [u_0, u_1]$  と表したとき,  $x := \langle G, \sigma \rangle$  を  $x$  に対する**束縛** (binding) と呼ぶ. 新たな TTSP 項グラフパターン  $g'$  は次のように  $g$  に対して束縛  $x := \langle G, \sigma \rangle$  を適用させることで得られる.

- (1)  $x$  でラベルづけされた  $g$  中の各変数  $h = [v_0, v_1] (v_0, v_1 \in V(g))$  に対して,  $H(g)$  から  $h$  を取り除く.
- (2)  $G$  のコピー  $G'$  を, 頂点  $v_0, v_1$  をそれぞれ頂点  $u'_0, u'_1 (u'_0, u'_1 \in V(G'))$  と同一視することで  $g$  に取り付ける ( $u'_0, u'_1$  はそれぞれ  $u_0, u_1$  に対応する).

このとき, 異なる変数ラベルに対する束縛の有限集合を**代入** (substitution)  $\theta$  と呼ぶ.  $g\theta$  は  $g$  に対してすべての束縛を同時に適用させることで得られる TTSP 項グラフパターンを表す.

内田ら [6] により提案された**形式グラフ体系** (Formal Graph System, 以下 FGS) に従い, TTSP 項グラフを項としてもつ論理プログラミングであるプリミティブ TTSP グラフ体系を次のように定義する.  $p$  を述語記号,  $g$  を TTSP 項グラフパターンとすると,  $p(g)$  を**アトム**という.  $A, B_1, \dots, B_n$  をアトムとする. このとき,  $A \leftarrow B_1, \dots, B_n (n \geq 0)$  の形をした節を**グラフ書換規則** (Graph Rewriting Rule) と呼び,  $A$  をヘッド部,  $B_1, \dots, B_n$  をボディ部という. 特に,  $n = 0$  のグラフ書換規則を**ファクト** (fact) という.  $o(g, x)$  を  $x$  でラベル付けされた変数が TTSP 項グラフパターン  $g$  に現れる回数とする.  $p(g) \leftarrow p_1(g_1), p_2(g_2), \dots, p_n(g_n) (n \geq 0)$  の形式で表現される以下の条件を満たすようなグラフ書換規則を**プリミティブ TTSP グラフ書換規則** (Primitive TTSP Graph Rewriting Rule, 以下  $pGRR_{TTSP}$ ) という.

- (1)  $n = 0$  つまりグラフ書換規則がファクトである場合,  $g$  はソースを表す頂点ラベル  $s$  を持つ頂点と, シンクを表す頂点ラベル  $t$  を持つ頂点とそれらに接続する1つの有向辺で構成される有向グラフである
- (2)  $n \geq 1$  の場合, 以下の条件を満たす
  - (a)  $g$  は  $|E(g)| + |H(g)| \geq 2$  である TTSP 項グラフパターンである
  - (b) すべての  $g_i (1 \leq i \leq n)$  は  $pGRR_{TTSP}$  である.
- (3) すべての変数  $x \in X$  に対して,  $o(g, x) = o(g_1, x) + \dots + o(g_n, x)$  かつ  $o(g, x) \leq 1$  である

**プリミティブ形式 TTSP グラフ体系プログラム** (primitive Formal TTSP Graph System Program,  $pFSTTSP$  プログラム) とは,  $pGRR_{TTSP}$  の有限集合をいう.  $pFSTTSP$  プログラムの例を図1に示す

グラフ書換規則  $\alpha = ' p(g) \leftarrow q_1(g_1), \dots, q_n(g_n) '$  に対し, ヘッド部とボディ部に現れる述語記号の集合をそれぞれ  $\Pi^h(\alpha)$  と  $\Pi^b(\alpha)$  とすると,  $\Pi^h(\alpha) = \{p\}$ ,  $\Pi^b(\alpha) = \{q_1, \dots, q_n\}$  と表せる.  $pFSTTSP$  プログラム  $\Gamma$

に対して,  $\Gamma$  中のすべての述語記号の集合は  $\Pi(\Gamma)$  と表記する. つまり,  $\Pi(\Gamma) = \bigcup_{\alpha \in \Gamma} \Pi^h(\alpha) \cup \Pi^b(\alpha)$  である. また任意の述語記号  $p \in \Pi(\Gamma)$  に対して, 述語記号が  $p$  である  $\Gamma$  中のファクトが持つすべての辺ラベルの集合は  $\Lambda(\Gamma, p)$  と表記される.

もし  $p$  と  $q$  が同じ述語記号で  $g_1\theta \simeq g_2\theta$  であれば, 代入  $\theta$  はアトム  $p(g_1)$  と  $q(g_2)$  の**単一化代入** (unifier) であると呼ばれる. 形式が  $\leftarrow B_1, \dots, B_m (m \geq 0)$  の規則を**ゴール**と呼ぶ. また  $m = 1$  のとき**ユニットゴール**,  $m = 0$  のとき**空ゴール**と呼ぶ. またグラフ書換規則  $\alpha$  に対して,  $var(\alpha)$  を  $\alpha$  中のすべての変数ラベルの集合として表記する.  $A_1$  と  $A_2$  を2つの項グラフパターンまたはアトムとする. ある代入  $\theta$  と  $\theta'$  に対し,  $A_1 \simeq A_2\theta$  かつ  $A_1\theta' \simeq A_2$  であるとき,  $A_1$  は  $A_2$  の**変種** (variant) であるという.

ここで,  $\Gamma$  を  $pFSTTSP$  プログラム,  $\beta$  をゴールとする.  $\Gamma$  を用いることで得られる  $\beta$  からの**導出**は以下の条件を満たす3個組  $(\beta_i, \theta_i, \alpha_i)$  である.

- (1)  $\beta_i$  はゴール,  $\theta_i$  は代入,  $\alpha_i$  は  $\Gamma$  中のグラフ書換規則の変種である.
- (2)  $\beta_0 = \beta$
- (3) 任意の  $i, j$  に対して  $var(\alpha_i) \cap var(\alpha_j) = \emptyset$  であり, また任意の  $i$  に対して  $var(\alpha_i) \cap var(\beta) = \emptyset$
- (4) ゴールのボディ部からアトムを選択する手続きを  $A$  とする.  $\beta_i$  をゴール  $\leftarrow A_1, \dots, A_k$ ,  $\alpha_i$  をグラフ書換規則  $A \leftarrow B_1, \dots, B_q$  とする.  $A_m (1 \leq m \leq k)$  を  $A$  によって選ばれる  $\beta_i$  のボディ部のアトムとする.  $\theta_i$  は  $A$  と  $A_m$  の単一化代入,  $\beta_{i+1}$  はゴール  $\leftarrow A_1\theta_i, \dots, A_{m-1}\theta_i, B_1\theta_i, \dots, B_q\theta_i, A_{m+1}\theta_i, \dots, A_k\theta_i$  である.

$\Gamma$  を  $pFSTTSP$  プログラム,  $\beta_0$  をゴールとする.  $\Gamma$  を用いて得られる**反駁**は, 空ゴールで終わるような,  $\Gamma$  を用いることで得られる  $\beta_0$  からの有限導出である. この時,  $F = (\beta_i, \theta_i, \alpha_i)_{0 \leq i \leq l}$  は,  $\Gamma$  を用いることで得られるユニットゴール  $\beta$  からの反駁を表す.  $pFSTTSP$  プログラム  $\Gamma$  と述語記号  $r$  に対し,  $\Gamma$  を用いてユニットゴール  $\leftarrow p(G)$  からの反駁が存在するすべての TTSP グラフ  $G$  の集合を  $pFSTTSP$  **言語**といい,  $L(\Gamma, r)$  で表す.

## 2.1 グラフニューラルネットワーク (GNN) モデル

深層学習モデルの1つに, グラフ構造データを入力として処理を行うことができる GNN(Graph Neural Networks) がある. GNN はグラフ構造データを扱えるため, ソーシャルグラフや化学の分子構造などの予測・解析に活用されている. GNN の中でも, 特徴量更新の手法によってさまざまなモデルが存在する. その一つに, グラフ畳み込みネットワーク GCN(Graph Convolutional Networks)[3] がある. GCN では, 各頂点の特徴ベクトルを持つグラフを入力とし, 頂点の特徴量の更新の際に, 隣接頂点の特徴量

を取り込む畳み込みを行うことで、頂点の特微量に隣接関係を反映できる。GCN のタスクには様々なものがあるが、本実験ではグラフのラベルを予測する二値グラフ分類を行う GCN を用いる。GCN は入力として 0 か 1, すなわち負例か正例かのラベルがついたグラフ集合を学習する。その後、ラベルが未知のグラフが与えられたときにそのグラフのラベルを予測する。GCN にも畳み込み処理の方法により様々な種類が存在する。本論文では、10 層の R-GCNs[7] からなるモデルと、3 層の GConvLSTM[8] からなるモデルの 2 つのモデルを構築し、質問学習アルゴリズムのオラクルとして用いる。以降は R-GCNs から構築されるモデルを RGCN, GConvLSTM から構築されるモデルを GConvLSTM と呼ぶことにする。

### 2.1.1 関係グラフ畳み込みネットワーク (RGCN) モデル

グラフの中でも頂点間の関係が存在する知識グラフが存在する。GCN の中でも、このような知識グラフを扱うことを目的としたモデルに R-GCNs(Relational Graph Convolutional Networks) がある。R-GCNs では、ある頂点の特微量を更新する際に、辺ラベルにより定義される重み行列を隣接頂点の特微量にかけ合わせる。これにより、辺ラベルの情報も考慮しつつ、隣接頂点の情報を取り込みながら頂点の特微量更新を行うことができる。第  $l+1$  層の頂点  $i$  に対する出力は以下のように定義される。

$$h_i^{(l+1)} = \sigma \left( \sum_{r \in R} \sum_{j \in N_r^l} \frac{1}{c_{i,r}} W_r^{(l)} h_j^{(l)} + W_0^{(l)} h_i^{(l)} \right) \quad (1)$$

ここで、 $h_i^{(l+1)}$  は第  $l+1$  層の頂点  $i$  の特徴ベクトルを表す。  $R$  はグラフの辺ラベルの集合を表しており、 $N_r^l$  は頂点  $i$  と辺ラベル  $r$  で接続する頂点の集合を表している。  $W_0^{(l)}$  は頂点  $i$  の特徴を更新するための重み行列で、 $W_r^{(l)}$  は辺ラベル  $r$  に対する重み行列である。  $c_{i,r}$  は正規化定数であり、 $\sigma$  は ReLU 関数などの活性化関数を表している。

### 2.1.2 長短期記憶ネットワーク (LSTM) モデル

GConvLSTM[8] は GCN とニューラルネットワークの 1 つである LSTM(Long Short-Term Memory) を組み合わせたものである。音声や文章などある要素が並べられたデータを時系列データという。LSTM は時系列データを対象として扱うモデルで、大きな特徴としてデータの長期依存関係を学習することが可能である。これにより、前のステップの要素の情報を保持することで、ある要素の情報を学習する際に過去のステップの情報を考慮して学習できる。GCN はレイヤを重ねるごとに離れた頂点の情報を取り込むため、比較的隣接関係の情報が強い最初のレイヤの情報は薄れていってしまう。しかし、LSTM の性質を GCN にかけて合わせることでこの問題を緩和できると考えられる。まず入力特微量に対して GCN により畳み込み処理を行う。ここは一般的な GCN のため R-GCNs と違い辺ラベルの情報は考慮しない。その後 LSTM での演算を通して

次の層の GCN に出力される。この演算で LSTM は一つ前の層の情報である  $h^{l-1}$  を用いて演算を行う。これにより過去の層の情報を考慮して特徴の学習を行うことが可能になる。

## 3. TTSP 項グラフパターン同定問題を解く質問学習アルゴリズム

常に正答を返す教師 (オラクル) に対して学習者が質問を繰り返し与えることでオラクルが持つ概念を同定する質問学習モデル [1] に基づいた TTSP 項グラフパターン同定問題を次のように定義する。

**設定:** 背景知識である FGS  $\Gamma$  と学習目標である  $pFS_TTSP$  言語  $L^*$

**インスタンス:** TTSP グラフ  $G \in L^*$

**問題:** 所属性質問を用いて、 $L^*$  と同一の  $pFS_TTSP$  言語を生成する FGS  $\Gamma \cup \{\alpha\}$  のルール  $\alpha$  を見つけよ。

本論文では、以下で定義される所属性質問について考える。  $pFS_TTSP$  プログラム  $\Gamma$ ,  $pGRR_TTSP \alpha$ ,  $\alpha$  のヘッド部の述語記号  $r$  に対して、学習目標を  $L^* = L(\Gamma \cup \{\alpha\}, r)$  としたとき、与えられる TTSP グラフ  $G$  が、 $L^*$  に含まれるとき "yes" を返し、含まれないときに "no" を返す。  $pFS_TTSP$  プログラムを用いた TTSP 項グラフパターン同定問題を解く質問学習アルゴリズムを以下に示す。

**入力:** 学習目標  $L^*$  に含まれる TTSP グラフ  $G$

**出力:**  $L^*$  と同一の言語を生成できる FGS  $\Gamma \cup \{\alpha\}$  のルール  $\alpha$

- (1)  $G$  に辺の縮約・削除を施しオラクルに質問する操作を再帰的に適用し  $L^*$  内の最小 TTSP グラフ  $G_{min}$  を求める
- (2) 各辺の辺ラベルを置き換えオラクルに質問する操作を施すことで  $G_{min}$  からルールの変数と述語を同定する
- (3) 2 で同定した変数と述語から  $L^*$  と同等の TTSP 項グラフパターンを生成する  $\Gamma \cup \{\alpha\}$  の  $\alpha$  を作成し出力する

本論文では、TTSP グラフの集合を学習した GCN モデルをオラクルとして用いる。並列操作に相当するルールのみ適用して得られる TTSP グラフは、ソースとシンク間の多重辺により構成される。多重辺には順序がないため、辺の数に着目した辺の削減に関する効率化が図れる。直列操作に相当するルールのみ適用して得られる TTSP グラフは、ソースとシンクを除く各頂点の入次数と出次数がそれぞれ 1 であるチェーン型の TTSP グラフである。辺ラベルが 1 つの場合、ソースからシンクまでの間の辺の数に着目

した、辺の縮約・削除の効率化が図れる。

辺の縮約・削除の効率化手法として、2分探索法を利用して次の操作を最小正例を獲得するまで行う。まず、質問学習アルゴリズムに入力された TTSP グラフ  $G$  に多重辺が含まれている場合、任意の2頂点間の多重辺を半分削除する。チェーン型の部分グラフは、ソースからシンクまでの間の辺の数を半分に減らして縮約する。縮約後の TTSP グラフ  $G'$  を入力としてオラクルに所属性質問を行う。 $G' \in L^*$  である場合は辺の縮約・削除を適用させ、同じ2頂点間に対し同様の操作を行う。 $G' \notin L^*$  である場合は縮約・削除前に戻して前回減らした辺の数の半分の辺を縮約・削除する。この操作を再帰的に適用する。

以降、効率化手法を用いていない質問学習アルゴリズムを  $VA^{TTSP}$  (Visualization Algorithm for TTSP graph), 多重辺の削除を効率化した手法を用いた質問学習アルゴリズムを  $ImpVA_P^{TTSP}$  (Improved VA for TTSP graph w.r.t Parallel operation), 多重辺の削除に加えチェーン型の部分グラフの縮約も効率化を行った質問学習アルゴリズムを  $ImpVA_P^{TTSP}$  (Improved VA for TTSP graph w.r.t Series and Parallel operations) と表記する。

## 4. 評価実験と考察

本章では、2種類の GCN をオラクルとした質問学習アルゴリズムを計算機上で実装し、人工的に作成した TTSP グラフのデータセットを用いて評価実験を行ったのでその報告を行う。プログラミング言語は Python3.13 で実装し、OS が Windows11 の計算機を 11 台用いた。

TTSP 項グラフパターン  $g$  とマッチする TTSP グラフの集合を  $S_+$ 、マッチしない TTSP グラフの集合を  $S_-$  とする。学習データセット  $S_+ \cup S_-$  を学習した高精度 GCN モデル  $M_{TTSP}$  をオラクルとする質問学習アルゴリズムによって獲得した TTSP 項グラフパターンのデータセットに対する二値分類精度である  $f$  値を次のように定義する。 $S_+ \cup S_-$  のなかで、質問学習アルゴリズムによって得られる TTSP 項グラフパターン  $g'$  とマッチする TTSP グラフの集合を  $S_+^G$ 、そうでないものを  $S_-^G$  と表すとき、 $f$  値は  $2 \cdot P \cdot R / (P + R)$  と定義される。ここで、 $P = |S_+ \cap S_+^G| / |S_+^G|$ 、 $R = |S \cap S_+^G| / |S_+|$  である。 $P$  は適合率 (Precision) を表し、正例と予測したものの中で、どれだけ正解であったかの割合を表している。 $R$  は再現率 (Recall) を表し、正例と予測したものの中で、どれだけ正例と予測できたかの割合を表している。 $f$  値はこれら2つの値の調和平均を取っている。

### 4.1 学習済み GNN の構築

人工データセットの TTSP グラフ  $G$  の各頂点の初期特徴量  $H^0$  は、以下の6次元のベクトル  $(h_1, h_2, h_3, h_4, h_5, h_6)$  で定義した。

$h_1$  : 頂点の入次数  
 $h_2$  : 頂点の出次数  
 $h_3$  : 頂点の隣接頂点数 (親)  
 $h_4$  : 頂点の隣接頂点数 (子)  
 $h_5$  : ソースから頂点までの最短距離 (ソースを 0 とする)  
 $h_6$  : 頂点からシンクまでの最短距離 (シンクを 0 とする)

RGCN と GConvLSTM の2つのモデルの内部構造について説明する。RGCN は、10層の R-GCNs 層、読み込み層、全結合層で構築した。初期特徴量  $H^0$  は (NumNode, 6) に設定されている。ここで、NumNode は学習する TTSP グラフの頂点数を表し、6 は特徴量のベクトル  $(h_1, h_2, h_3, h_4, h_5, h_6)$  の次元数を表す。複数の R-GCNs 層の畳み込みと活性化関数である ReLU による処理を通して特徴量を更新する。その後、読み込み層ですべての頂点の特徴量をまとめることでグラフの特徴量を得る。まとめる方法にも様々な種類があるが今回は各特徴量の和を取る add pool を用いる。グラフの二値分類タスクを行うモデルのため、全結合層である Linear で出力を2次元のベクトルとする。最後に sigmoid 関数を適用し出力としてベクトル  $o = (o_1, o_2)$  ( $0 \leq o_1, o_2, \leq 1$ ) を得る。モデルの学習の際は、損失関数として二値交差エントロピー、最適化アルゴリズムは Adam を用い、学習率を  $10^{-2}$  に設定した。加えて学習には early stopping を用いた。これにより、モデルの性能が向上しなくなるタイミングで学習をやめることで、モデルの過学習を抑えることが期待できる。

GConvLSTM は、3層の GConvLSTM 層、読み込み層、全結合層で構築した。初期特徴量  $H^0$  は (NumNode, 6) となっており、複数の GConvLSTM 層と ReLU による処理を通して特徴量を更新する。そして、読み込み層ですべての頂点の特徴量をまとめる。RGCN とは異なり GConvLSTM では各特徴量の平均をとる add mean を用いた。その後は全結合層である Linear で出力を2次元にしたのち、sigmoid 関数を適用することで出力ベクトルを得る。学習の際のパラメータは RGCN 同様、損失関数として二値交差エントロピー、最適化アルゴリズムは Adam を用い、学習率を  $10^{-2}$  に設定した。加えて過学習を抑えるため early stopping を用いて学習を行った。

### 4.2 評価実験 1 とその考察

質問学習アルゴリズムのオラクルとして用いる2つの深層学習モデルの性能比較を行った。使用した GCN モデルは GConvLSTM と RGCN である。辺ラベル集合  $\Lambda = \{a\}$  に基づき、データセット1個につき TTSP 項グラフパターンとマッチする TTSP グラフを正例として 3000 個、ランダムに生成した TTSP グラフから TTSP 項グラフパターンとマッチしないものを負例として 5000 個用意した。訓練データを 5120 個、検証データを 1280 個、テストデータを 1600 個に分けて GCN モデルに学習させた。

学習目標となる TTSP 項グラフパターンの変数に指定の TTSP グラフを代入することで得られる正例を、質問学習アルゴリズムに1つ入力し、これをデータセット 1089 個分を行った。変数に代入する TTSP グラフを変えて実験を行った。1 回目は 20 本の多重辺を直列に 2 つ連結した TTSP グラフ、2 回目は 20 本の多重辺、3 回目は頂点数 20 個のチェーン型のグラフを代入した。代入したグラフを図 2 に示す。

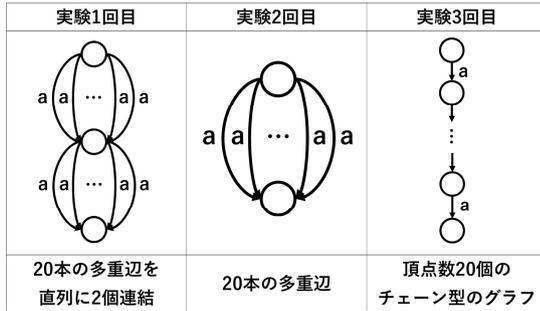


図 2: 学習目標となる TTSP 項グラフパターンに代入する TTSP グラフ

「GCN モデルによって f 値の平均値に差はない」という帰無仮説の下で、ウェルチの法による t 検定を用いて使用した GCN モデルによる f 値の差を比較した結果が以下の表 1~9 である。

表 1:  $VA^{TTSP}$  の f 値の比較 (実験 1 回目)

	GConvLSTM	RGCN
平均	0.158	0.185
分散	0.105	0.13
t	1.866	
p(両側)	0.062	

表 2:  $ImpVA_P^{TTSP}$  の f 値の比較 (実験 1 回目)

	GConvLSTM	RGCN
平均	0.229	0.162
分散	0.14	0.103
t	4.45	
p(両側)	$p < 0.05$	

表 3:  $ImpVA_{SP}^{TTSP}$  の f 値の比較 (実験 1 回目)

	GConvLSTM	RGCN
平均	0.158	0.133
分散	0.112	0.088
t	1.852	
p(両側)	0.064	

表 4:  $VA^{TTSP}$  の f 値の比較 (実験 2 回目)

	GConvLSTM	RGCN
平均	0.213	0.135
分散	0.144	0.1003
t	5.236	
p(両側)	$p < 0.05$	

表 5:  $ImpVA_P^{TTSP}$  の f 値の比較 (実験 2 回目)

	GConvLSTM	RGCN
平均	0.288	0.179
分散	0.175	0.125
t	6.59	
p(両側)	$p < 0.05$	

表 6:  $ImpVA_{SP}^{TTSP}$  の f 値の比較 (実験 2 回目)

	GConvLSTM	RGCN
平均	0.133	0.107
分散	0.102	0.09
t	1.953	
p(両側)	0.051	

表 7:  $VA^{TTSP}$  の f 値の比較 (実験 3 回目)

	GConvLSTM	RGCN
平均	0.0944	0.108
分散	0.074	0.073
t	1.164	
p(両側)	0.122	

表 8:  $ImpVA_P^{TTSP}$  の f 値の比較 (実験 3 回目)

	GConvLSTM	RGCN
平均	0.093	0.107
分散	0.072	0.074
t	1.206	
p(両側)	0.114	

表 9:  $ImpVA_{SP}^{TTSP}$  の f 値の比較 (実験 3 回目)

	GConvLSTM	RGCN
平均	0.086	0.074
分散	0.069	0.056
t	1.131	
p(両側)	0.129	

表 2 より、実験 1 回目の  $ImpVA_P^{TTSP}$  において GConvLSTM と RGCN で f 値の平均値に有意な差があり、f 値の平

均値は GConvLSTM が高い。また表 5 より、実験 2 回目においても  $ImpVA_P^{TTSP}$  において GConvLSTM と RGCN で  $f$  値の平均値に有意な差が認められ、 $f$  値の平均値は GConvLSTM が高い。

モデル間で  $f$  値に差が出た要因に対する考察として、RGCN は辺ラベルを考慮することが可能だが、今回は辺ラベルが 1 種類のデータセットを作成したため辺ラベルによる特徴の差が出ず、学習において影響が少なかったと考えられる。一方で GConvLSTM は、LSTM の効果により頂点間の隣接関係の特徴をより深く学習できるため、次数などの頂点の特徴量を捉えることができたのではないかと考えられる。

### 4.3 評価実験 2

既知の正例に対する質問学習アルゴリズムの可視化精度の検証を行った。データセット 1 つにつきグラフパターンとマッチするグラフを正例として 3000 個、ランダムに生成したグラフからグラフパターンとマッチしないものを負例として 5000 個用意し、GCN モデルに学習させる。使用したモデルは GConvLSTM である。質問学習アルゴリズムには、GCN モデルに学習させた正例 3000 個の内 2000 個を入力する。2000 個のデータの中で  $f$  値が最大のものをそのデータセットの代表値として用いる。この時用意したデータセットは 548 個である。

各質問学習アルゴリズムの質問回数を  $t$  検定 (ウェルチの法) を用いて比較した結果を表 10、表 11 に示す。

表 10:  $VA^{TTSP}$  と  $ImpVA_P^{TTSP}$  の質問回数の比較

	$VA^{TTSP}$	$ImpVA_P^{TTSP}$
平均	78.206	46.491
分散	4140.069	1150.024
t	10.208	
p(両側)	$p < 0.05$	

表 11:  $ImpVA_P^{TTSP}$  と  $ImpVA_{SP}^{TTSP}$  の質問回数の比較

	$ImpVA_P^{TTSP}$	$ImpVA_{SP}^{TTSP}$
平均	46.491	36.677
分散	1150.024	983.064
t	4.974	
p(両側)	0.525869	

「 $VA^{TTSP}$  と  $ImpVA_P^{TTSP}$  の質問回数の平均値に差はない」という帰無仮説を立てたとき、表 10 より  $VA^{TTSP}$  と  $ImpVA_P^{TTSP}$  の質問回数の平均値に有意な差があり、 $ImpVA_P^{TTSP}$  が  $VA^{TTSP}$  より質問回数の平均値が低い。また「 $ImpVA_P^{TTSP}$  と  $ImpVA_{SP}^{TTSP}$  の質問回数の平均値に差はない」という帰無仮説を立てたとき、表 11 より

$ImpVA_P^{TTSP}$  と  $ImpVA_{SP}^{TTSP}$  の質問回数の平均値に有意な差があり、 $ImpVA_{SP}^{TTSP}$  が  $ImpVA_P^{TTSP}$  より質問回数の平均値が低い。表 10、表 11 の結果より、効率化の対象を多重辺の削除、チェーン型の部分グラフの縮約と広げるほど質問回数が減少していることが分かる。

また各質問学習アルゴリズムの  $f$  値を  $t$  検定 (ウェルチの法) を用いて比較した結果を表 12、表 13 に示す。

表 12:  $VA^{TTSP}$  と  $ImpVA_P^{TTSP}$  の  $f$  値の比較

	$VA^{TTSP}$	$ImpVA_P^{TTSP}$
平均	0.575	0.631
分散	0.166	0.156
t	2.335	
p(両側)	$p < 0.05$	

表 13:  $ImpVA_P^{TTSP}$  と  $ImpVA_{SP}^{TTSP}$  の  $f$  値の比較

	$ImpVA_P^{TTSP}$	$ImpVA_{SP}^{TTSP}$
平均	0.631	0.616
分散	0.156	0.166
t	0.635	
p(両側)	0.526	

「 $VA^{TTSP}$  と  $ImpVA_P^{TTSP}$  の  $f$  値の平均値に差はない」という帰無仮説を立てたとき、表 12 より  $VA^{TTSP}$  と  $ImpVA_P^{TTSP}$  の  $f$  値の平均値に有意な差があり、 $ImpVA_P^{TTSP}$  が  $VA^{TTSP}$  より  $f$  値の平均値が高い。また「 $ImpVA_P^{TTSP}$  と  $ImpVA_{SP}^{TTSP}$  の  $f$  値の平均値に差はない」という帰無仮説を立てたとき、表 13 より  $ImpVA_P^{TTSP}$  と  $ImpVA_{SP}^{TTSP}$  の  $f$  値の平均値に有意な差は認められない。

3 つの質問学習アルゴリズムの  $f$  値に差が生じた要因に対する考察として、GCN をオラクルとした場合、誤った回答を返す可能性があるため、効率化手法を用いることで質問回数が減ったことにより、オラクルが誤った回答をする回数を減らすことで  $f$  値が向上したのではないかと考える。特に多重辺の削除による各頂点の特徴量の変化は削除した辺のソースとシンクのみと局所的であることが可視化精度が向上する要因であると考えられる。また多重辺に対して 1 本ずつ辺の削除をした際、一度負例と判断すると以降同じ多重辺に対して同様の削除操作が続き、縮約後の形が同じ TTSP グラフがオラクルに入力されるため負例として判断されその多重辺はそれ以上辺を削除されない。そのため誤って負例と判定された場合最小正例の獲得の精度が低下する。多重辺の削除を効率化することで、同じ多重辺に対して削除する辺の本数が変わることで誤って負例と判断された場合でもその後正しく正例を判断される可能性が高まると考えられる。

また質問学習アルゴリズムから出力された TTSP 項グラ

フパターンの例を図3に示す。

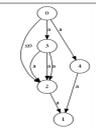
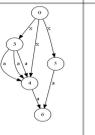
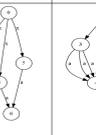
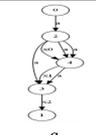
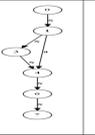
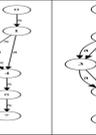
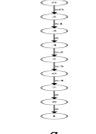
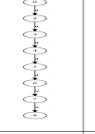
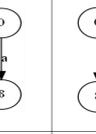
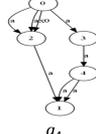
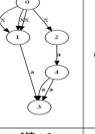
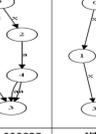
学習目標となる TTSP項グラフパターン	$VA^{TTSP}$	$ImpVA_P^{TTSP}$	$ImpVA_{SP}^{TTSP}$
 $g_1$	 f値: 1	 f値: 1	 f値: 1
 $g_2$	 f値: 0.997175	 f値: 0.970403	 f値: 0.781861
 $g_3$	 f値: 0.044285	 f値: 0	 f値: 0
 $g_4$	 f値: 1	 f値: 0.998835	 f値: 0

図3: 出力されたTTSP項グラフパターンの例

図3の $g_2$ と $g_4$ のような多重辺を含むTTSP項グラフパターンにおいて、特に $ImpVA_P^{TTSP}$ と $ImpVA_{SP}^{TTSP}$ の多重辺の本数が正確に抽出されていない傾向がみられた。また $g_3$ のように、チェーン型の構造を持つTTSP項グラフパターンにおいてはf値が低い傾向がみられた。

## 5. おわりに

本研究では、質問学習モデルに基づいたTTSP項グラフパターン同定問題を解く質問学習アルゴリズムを提案し、質問回数に関する効率化について評価実験を行い、質問回数の効率化及びアルゴリズムの精度向上を確認できた。今後の課題として、TTSPグラフの構造的特徴と可視化精度との関連について、各頂点の次数や各頂点におけるソース及びシンクからの距離といった特徴量を用いてより詳細に解析する必要があると考えられる。

**謝辞** 本研究はJSPS科研費JP22K12172, JP24K15074, JP24K15090の助成を受けたものです。

## 参考文献

[1] D. Angluin: Queries and Concept Learning, Machine Learning, 2(4):319-342(1988).  
 [2] T. Uchida, et al.: Exact Learning of Primitive Formal System Defining Labeled Ordered Tree Language via Queries, IEICE Trans. Inf. Syst., E102-D,

(3):470-482 (2019).

[3] Thomas, N. Kipf, Welling, M.: Semi-Supervised Classification with Graph Convolutional Networks, arXiv:1609.02907, 22 Feb 2017, <https://arxiv.org/abs/1609.02907>, accessed 6 Jan 2026.  
 [4] 石灘 洗樹, 正代 隆義, 内田 智之, 松本 哲志: 超高精度グラフ畳み込みネットワークをオラクルとする無順序木パターンの質問学習モデル, 情報処理学会第85回全国大会, 4M-06 (2022).  
 [5] 東山 的生, 内田 智之, 正代 隆義, 松本 哲志: 順序項木パターンに対する学習済GCNをオラクルとした質問学習アルゴリズムのランダム化による精度向上, 火の国情報シンポジウム 2024, B-6-1 (2024).  
 [6] T.Uchida, T.Shoudai, and S.Miyano : Parallel algorithms for refutation tree problem on formal graph systems, IEICE Trans. Inf.Syst., E78-D, (2):99-112 (1995).  
 [7] Schlichtkrull, M., Thomas, N. Kipf, Bloem, P., Berg, R. van den, Titov, I., Welling, M.: Modeling Relational Data with Graph Convolutional Networks, in Proc.ESWC 2018, :The Semantic Web, pp.593-607 (2018).  
 [8] Seo, Y., Defferrard, M., Vandergheynst, P., Bresson, X.: Structured Sequence Modeling with Graph Convolutional Recurrent Networks, Neural Information Processing, in Proc.ICONIP 2018, pp.362-373 (2018).