

深層学習を用いた未知ノードの出現を考慮した時系列グラフの予測

山崎 翔平[†] 原田 圭[†] 佐々木勇和[†] 鬼塚 真[†]

[†] 大阪大学大学院情報科学研究科 〒 565-0871 大阪府吹田市山田丘 1-5

E-mail: †{yamasaki.shohei,harada.kei,sasaki,onizuka}@ist.osaka-u.ac.jp

あらまし グラフ構造は現実世界の複雑な現象をモデル化する一般的なデータ表現であり、時間的な変化を伴うことが多い。時系列グラフを対象とした属性予測やリンク予測では、新たに出現する未知のノードや消失するノードについては考慮していない。本稿では、未知のノードを含む時系列グラフの将来のノード集合、リンク集合、全てのノードが持つ属性を同時に予測する問題について取り組む。そこで、属性予測とリンク予測を時系列グラフ上に拡張した新たな深層学習フレームワークを提案する。提案フレームワークは Graph Neural Network (GNN) と Gated Convolutional Neural Network (Gated CNN) を統合した GNN ブロックを有しており、時系列グラフの属性変化とトポロジ変化を同時に考慮した将来予測を実現する。実世界のデータセットを用いて提案フレームワークによって予測したグラフの精度の検証を行う。

キーワード network embedding, リンク予測, 属性予測

1. はじめに

グラフ構造は現実世界の複雑な現象をモデル化する一般的なデータ表現である。グラフ構造で表現されるデータとしては論文の引用ネットワークやソーシャルネットワークが挙げられる。実世界に存在するグラフ構造の多くはノードが属性値を有し、また時間変化によってグラフ構造が変化する。このような時間変化を伴う属性付きのグラフは、属性付き時系列グラフと呼ばれる。例えば、ソーシャルネットワークの場合、ノードをユーザ、リンクを友人関係、属性値をプロフィールとすればユーザ数、友人関係、プロフィールは時々刻々と変化するため、属性付き時系列グラフとしてモデル化できる。SNS や IoT の普及により属性付き時系列グラフとしてモデル化できる対象はますます増加傾向にあり、その応用の一つとして属性付き時系列グラフを対象とした予測技術が注目を集めている。特に、SNS は個人利用に留まらず企業や組織の広報としての利用も増えており、過去の情報から将来のユーザ間の友人関係やプロフィール情報を予測することはマーケティング活動に有効な施策である。

時間変化する属性付き時系列グラフの将来の構造を予測する技術は多様な分野への応用が期待できるとされ大きな注目を集めている。特に、将来のグラフにおけるリンク関係を予測するリンク予測や、ノード属性を予測する属性予測は実世界での利用性の高さから多くの技術が提案されている [1-3]。リンク予測とは過去から現在までの一定期間における時系列グラフを用いて、将来のグラフが持つリンク関係を予測するタスクである。一方で、属性予測とは過去から現在までの一定期間における属性付きグラフを用いて、将来のノードが有する属性を予測するタスクである。これまでの研究ではリンク予測と属性予測は別々に研究されており、属性付き時系列グラフに対して属性予測とリンク予測を同時に行った研究はない。既存研究ではノードあるいはリンクの不変性を仮定することでリンクあるいは属性予測問題に帰着させているためである。ノードおよびリンク

の両方の時間変化を仮定した場合、互いの変化に影響し合うため既存の時系列データ予測手法では予測することはできない。また、これらのタスクでは共通して過去に観測されたノードのみを予測の対象としており、将来新たに出現する未知のノードについては考慮しておらず、新たに出現する未知ノードに対して属性またはリンク関係を予測することは簡単ではない。

そこで、本稿では属性付き時系列グラフに対してノードおよびリンクの両方の時間変化を仮定し、将来のグラフに未知ノードの出現を考慮したグラフ予測問題について取り組む。この問題を属性付き時系列グラフ予測問題と呼ぶ。属性付き時系列グラフ予測問題では将来のグラフにおけるリンク予測と属性予測を同時に予測する。加えて将来のグラフには過去に観測していない未知ノードの出現を考慮するため、ノード予測も同時に行う。実世界に存在する属性付き時系列グラフの多くはノード変化、リンク変化、および属性変化が同時に発生し、さらに将来のグラフにおいて未知ノードが出現する可能性も十分にある。そのため、属性付き時系列グラフの予測問題を解くことで既存のリンク予測や属性予測と比較してより多くの分野へ応用できることが期待される。

しかしながら、属性付き時系列グラフ予測問題を解くことは容易ではなく、筆者らの知る限り既存研究も存在しない。互いに影響し合うグラフのノード変化、リンク変化、および属性変化を考慮し、また将来のグラフに未知ノードが出現する可能性も考慮しながらグラフの全体像を予測する必要がある。そこで、属性付き時系列グラフ予測問題を解くための深層学習フレームワークを新たに提案する。提案フレームワークは属性付き時系列グラフに対して、将来のグラフにおける未知ノードの出現を考慮しながらノード、リンク、および属性を同時に予測する。また、グラフのトポロジカルな特徴を学習する Graph Neural Networks (GNNs) を応用したアプローチを採用することで高精度な予測を実現する。提案フレームワークはグラフの空間的な特徴をモデル化する GNNs 層と、時間的な特徴をモデ

ル化するための Gated Convolutional Neural Network(Gated CNN) 層 [4] の組み合わせからなる。特に, GNNs 層では Gated Graph Neural Network (GGNN) [5] を採用することでグラフの空間的な変化をグラフの時間的な変化に対応させる。実験では, 実世界のデータセットとして NBA 選手のソーシャルネットワークを用いて提案フレームワークの評価を行い, ノード予測, リンク予測, 属性予測のすべてにおいて高精度な予測結果を得られることを示す。

本稿の構成は以下の通りである。2 章にて関連研究について説明し, 3 章にて事前知識について説明する。4 章にて属性付き時系列グラフ予測問題を定義し, 5 章にて提案手法フレームワークについて説明する。6 章にて評価実験について述べ, 7 章で本稿をまとめる。

2. 関連研究

以下では関連研究としてグラフの属性予測とリンク予測について説明する。

属性予測は, 予測の長さに基づいて短期, 中長期の 2 つのスケールに分類される。属性予測の最も単純な統計的アプローチ (例えば, 線形回帰) は, 短期では上手くいくことが知られているが, グラフ構造の不確実性, 複雑さのためにこの方法は比較的長期の予測にはあまり有効でない [1]。中長期的な属性予測に対するアプローチとしては, 時系列分析手法である Autoregressive Integrated Moving Average (ARIMA) や, Support Vector Regression (SVR), Neural Networks (NN) などの機械学習手法が挙げられる。特に, Deep Belief Network (DBN) や Stacked Autoencoder (SAE) 等の深層学習を用いたアプローチが適用されるようになり, 属性予測の精度は大幅に向上した [6–8]。しかしながら, DBN や SAE 等の Deep Neural Network (DNN) モデルが入力から空間的および時間的特徴を共同して抽出することは困難である。そこで, 空間的特徴を抽出するために, Convolutional Neural Network (CNN) を利用することでノード間の隣接関係を捕捉し, 時間的特徴を抽出するために Recurrent Neural Network (RNN) を利用することでグラフ構造の時間変化を捕捉する手法が台頭した。Wu と Tan は 1-D CNN と Long Short-Term Memory (LSTM) [9] を組み合わせることで, 空間的及び時間的特徴を抽出しながら属性予測を行うフレームワークを提案した [10]。しかし, ここで適用される畳み込み演算はグリッド構造 (例えば画像, ビデオ) に適用されるものでありグラフ構造に最適な演算処理ではない。近年, グラフ構造に対する畳み込み演算を行う NN モデルである Graph CNN に関する研究 [11, 12] が盛んに行われており, Graph CNN を応用したグラフ構造の属性予測フレームワークとして Spatio-Temporal Graph Convolutional Network (STGCN) が提案された [1]。STGCN はグラフ構造の空間的特徴を抽出するために Graph CNN を, 時間的特徴を抽出するために 1-D CNN を利用しており, 属性予測問題の一つである交通量予測において既存手法を上回る精度を達成している。

リンク予測は静的リンク予測と動的リンク予測に分けられる。

静的リンク予測はある時点でのタイムステップを 1 つだけ考慮する。すなわち, 時間変化を含まない静的なネットワークにおいて, 周辺ノードの属性や周囲のリンクの有無等, グラフの構造的な情報のみから潜在的なリンクを推測する [13]。動的リンク予測は, 過去の一連のタイムステップで観測されたグラフ構造に基づいて将来の潜在的なリンクを推測する予測を指す [14]。本稿では動的リンク予測を行うため, 以降で単にリンク予測と述べる場合には, 動的リンク予測を意図しているものとする。リンク予測手法としては, SVD 等の行列分解を用いたアプローチや, Adamic/Adar 等の共通隣接ノードに関する構造類似度を用いてランク付けを行う方法がある [14, 15]。また, 多くの場合, グラフ構造のリンク関係はノード属性に強く依存することが知られており, ノード属性を活用することでリンク予測の性能を向上させる研究も行われている [16]。しかしながら, これらの手法ではグラフ構造が既知であり, 全ての隣接情報を利用できる状況にある場合を仮定している。本稿で取り組む属性付き時系列グラフ予測問題では, ノード変化と属性変化が同時に発生し, さらに未知ノードの出現も考慮しなければならない。このような状況におけるリンク予測手法については, これまでのところ詳しい議論は行われていない。

3. 事前知識

提案手法では属性予測の精度を向上させるためにグラフの空間的および時間的特徴を利用する。空間的特徴を抽出するために GNNs を, 時間的特徴を抽出するために Gated CNN をそれぞれ利用する。特に, GNNs では高精度なモデルである GGNN を利用する。本章では, はじめに本稿で用いる表記の定義を行う。次に GGNN および Gated CNN についてそれぞれ説明を行う。

3.1 表記の定義

本稿で取り組む属性付き時系列グラフ予測問題では各ノードが属性を持つグラフを対象としている。属性とはノードに関連付けられた情報でノードの特徴量として表現される。例えば, ソーシャルネットワークの場合, 人物のプロフィール情報がノード属性に対応する。また, 本稿では時間変化するグラフをノード, リンク, ノード属性が時間変化するグラフと定義する。時間変化にはこれまで観測していない新たなノード/リンク/属性が出現する場合, 現在は存在しないが過去に存在していたノード/リンク/属性が再出現する場合, そして存在するノード/リンク/属性が消失する場合がある。以下に詳細な定義を行う。

定義 1 (属性付きグラフ) ノードの集合 \mathcal{V} , 2 つのノード間を結ぶリンクの集合 \mathcal{E} , および各ノードの属性集合 $X = \langle x^1, x^2, \dots, x^{|\mathcal{V}|} \rangle \in \mathbb{R}^{|\mathcal{V}| \times n}$ の三つ組 $\mathcal{G} := (\mathcal{V}, \mathcal{E}, X)$ を属性付きグラフと呼ぶ。

定義 2 (属性付き時系列グラフ) 属性付きグラフ \mathcal{G} のノード集合 \mathcal{V} , リンク集合 \mathcal{E} , 属性集合 X , またはそのどれかが時間によって変化する場合, \mathcal{G} を属性付き時系列グラフと呼ぶ。またこのとき, タイムステップ t における属性付き時系列グラフ

を $\mathcal{G}_t := (\mathcal{V}_t, \mathcal{E}_t, X_t)$ と表記する.

定義 3 (属性付き時系列グラフの累積ノード集合) 属性付き時系列グラフを \mathcal{G}_t とする. タイムステップ t から L ステップ前までに観測された属性付き時系列グラフのノード集合を $\mathcal{U}_t(L) := \bigcup_{i=t-L+1}^t \mathcal{V}_i$ とする.

定義 4 (属性付き時系列グラフの累積リンク集合) 属性付き時系列グラフを \mathcal{G}_t とする. タイムステップ t から L ステップ前までに観測された属性付き時系列グラフのリンク集合を $\mathcal{D}_t(L) := \bigcup_{i=t-L+1}^t \mathcal{E}_i$ とする.

定義 5 (滞在ノード) タイムステップ $t+1$ において, $v \in \mathcal{V}_{t+1}$ かつ $v \in \mathcal{V}_t$ であるならば, v を滞在ノードと呼ぶ.

定義 6 (消失ノード) タイムステップ $t+1$ において, $v \notin \mathcal{V}_{t+1}$ かつ $v \in \mathcal{V}_t$ であるならば, v を消失ノードと呼ぶ.

定義 7 (未知ノード) タイムステップ $t+1$ において, $\mathcal{V}_{t+1} \setminus \mathcal{U}_t(L) \neq \emptyset$ であるとき, ノード $v \in \mathcal{V}_{t+1} \setminus \mathcal{U}_t(L)$ を未知ノードと呼ぶ.

定義 8 (復帰ノード) タイムステップ $t+1$ において, $v \in \mathcal{V}_{t+1}$ かつ $v \notin \mathcal{V}_t$ であるとき, v が未知ノードでないならば, v を復帰ノードと呼ぶ.

3.2 Graph Neural Networks

GNNs はグラフ構造を入力とした NN であり, Spatial method と Spectral method [11, 12] の大きく 2 つの手法に分けられる. 本稿ではモデルの改変のしやすさと, トポロジの時間変化に対応させるために Spatial method を用いる.

Li らは, Gated Recurrent Unit (GRU) [17] を用いた Spatial method の GNNs モデルである GGNN [5] を提案している. グラフ \mathcal{G} のノード v について, v に近接するノード集合を $N(v)$ で表現する. GGNN は隣接ノードの表現ベクトル h_w , $w \in N(v)$ からメッセージ m_v を形成し, v の次ステップの表現ベクトルを m_v を用いて更新する. 更新は式 (1) と式 (2) を繰り返して行う.

$$m_v^{s+1} = \sum_{w \in N(v)} h_w^s A_{e_{vw}} \quad (1)$$

ここで, h_w^s はノード w の更新 s ステップ目の表現ベクトルであり, S ステップまで更新を行う. e_{vw} はノード v , w 間のリンクのラベルを表している. $A_{e_{vw}}$ はリンクのラベルごとに割り当てられる重みである. GRU によりノードの表現ベクトルを更新する.

$$h_v^{s+1} = \text{GRU}([h_w^s; m_v^{s+1}]) \quad (2)$$

[...;...] はベクトルの連結を表す.

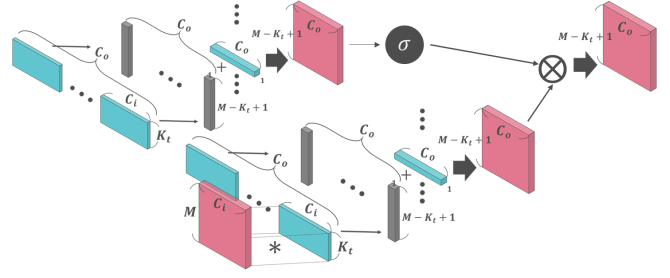


図1 Gated CNN のアーキテクチャ

3.3 Gated Convolutional Neural Network

時系列データの予測問題では RNN ベースのモデルや CNN が一般的に利用される. RNN ベースのモデルは時系列データを取り扱うことができ比較的高精度な予測を実現できる. 一方で, RNN ベースのモデルは CNN と比較してモデルが複雑であるため学習に長時間を要するという欠点もある. そこで, RNN ベースのモデルの一つである LSTM に利用されているゲーティング機構を CNN に取り入れた Gated CNN が提案された [4]. Gated CNN は高い予測精度と高速な学習を実現できる. 図1に Gated CNN のアーキテクチャを示す. Gated CNN はカーネルサイズ K_t を伴う 1-D CNN と非線形性のゲーティング機構 (GLU) を含む. 1-D CNN はパディングを行わない場合, 畳込み処理毎に $K_t - 1$ だけ入力要素は短くなる. 入力時系列データの長さを M , 入力チャンネル数を C_i , 出力チャンネル数を C_o とするとき, 入力時系列データ $Y \in \mathbb{R}^{M \times C_i}$ は畳込みカーネル $\Gamma \in \mathbb{R}^{K_t \times C_i \times 2C_o}$ によって GLU 内のゲート入力 $[PQ] \in \mathbb{R}^{(M-K_t+1) \times 2C_o}$ へ写像される. したがって, Gated CNN における入力の畳込み演算は以下のように定義される.

$$\Gamma * Y = P \otimes \sigma(Q) \quad (3)$$

ここで $*$ は畳込み演算子, \otimes は要素積, $\sigma(Q)$ は上層へ送る情報を制御するシグモイドゲートをそれぞれ表す. なお, $P \otimes \sigma(Q) \in \mathbb{R}^{(M-K_t+1) \times C_o}$ である.

4. 属性付き時系列グラフ予測問題

この章では, 本稿で取り組む属性付き時系列グラフ予測問題を定義する.

問題定義 (属性付き時系列グラフ予測問題) タイムステップ t から L ステップ前までの一連の属性付き時系列グラフ $\{\mathcal{G}_{t-L+1}, \mathcal{G}_{t-L+2}, \dots, \mathcal{G}_t\}$ が与えられたとき, タイムステップ $t+1$ における属性付き時系列グラフ \mathcal{G}_{t+1} を予測する問題を属性付き時系列グラフ予測問題と呼ぶ.

属性付き時系列グラフ予測問題では予測する対象としてノード集合, リンク集合, 属性集合の三つが考えられる. 以下に各対象の予測問題について説明する.

(1) ノード集合予測 : タイムステップ $t+1$ におけるノード集合 \mathcal{V}_{t+1} を予測する. ノード集合の予測には, まず \mathcal{V}_{t+1} に含まれているノード数を予測する必要がある. その後, 未知ノード

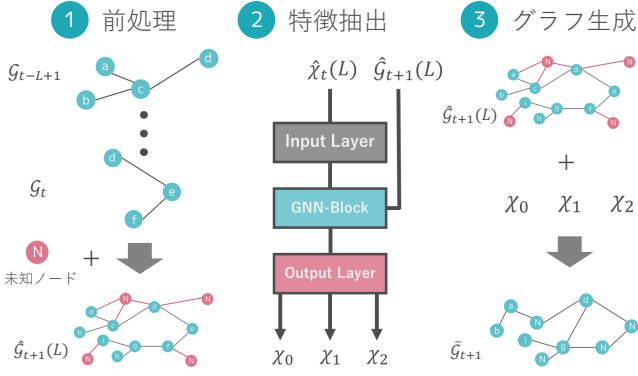


図 2 提案フレームワーク概要

ド、復帰ノード、および消失ノードを特定することにより \mathcal{V}_{t+1} を予測する。

(2) リンク集合予測 : タイムステップ $t+1$ におけるリンク集合 \mathcal{E}_{t+1} を予測する。リンク集合の予測には、まず観測された時系列グラフから、リンクの有無について影響の大きい属性とリンクが存在するノード間の類似度を決定する必要がある。その後、タイムステップ $t+1$ におけるノード間の属性を用いることによりリンクの有無を予測する。

(3) 属性集合予測 : タイムステップ $t+1$ における属性集合 X_{t+1} を予測する。属性集合の予測には、タイムステップ t から L ステップ前までに観測された属性付き時系列グラフにおけるノードの属性変化とタイムステップ $t+1$ におけるリンクの関係性を利用する。

ここで、ノード属性の一部は隣接するノードから影響されること、およびリンクの有無は隣接するノード間の一部の属性の類似度により決定されると仮定する。すなわち、ノード集合、リンク集合、および属性集合予測は互いに独立ではなくそれぞれの予測が他の予測に影響する。例えば、ノード集合予測では消失ノードを予測し、これは属性集合予測に影響する。また、属性集合予測で予測するノード属性はリンク集合予測におけるノード間類似度の決定に影響する。一方で、リンク集合予測で予測するリンク関係は属性集合予測における属性決定に影響する。したがって、属性付き時系列グラフに対してノード集合、リンク集合、属性集合を同時に予測する必要がある。

5. 提案フレームワーク

この章では属性付き時系列グラフ予測問題に取り組むフレームワークについて詳細を述べる。図 2 に提案フレームワークの概要を示す。まず、前処理にて、属性付き時系列グラフの時空間相関を補足する特徴抽出アーキテクチャへの入力グラフを生成する (5.1 節)。次に、特徴抽出アーキテクチャにて属性予測値、復帰ノードと消失ノードの確率を出力する (5.2 節)。最後に得られた特徴を基に予測結果となる出力グラフを生成する (5.3 節)。

5.1 前処理

特徴抽出アーキテクチャへの入力とするグラフを生成する。入力グラフは観測された時系列グラフのノード集合とリンク

集合、タイムステップ $t+1$ で出現を予測されたノード集合とリンク集合を全て含んでいる。これらの集合はタイムステップ $t+1$ におけるノード集合 \mathcal{V}_{t+1} とリンク集合 \mathcal{E}_{t+1} の候補となり得るもので、特徴抽出アーキテクチャはこの候補の中から予測結果となる出力グラフを特定する。また、特徴抽出アーキテクチャには入力グラフが保持するノードの属性が入力され、将来のノードの属性 X_{t+1} の予測結果を出力する。以下で、入力グラフの生成方法と、これが保持するノードの属性について詳細を述べる。

5.1.1 主要な属性の決定

リンクの有無は、隣接するノード間の一部の属性の類似度により決まる。ここでは、リンクの有無について最も大きな影響を与える属性を主要な属性と呼ぶ。主要な属性は以下のように、 \cos 類似度を用いて決定される。

$$\exists p \in \arg \max_j \sum_{t'} \sum_{(v_1, v_2) \in \mathcal{E}_t} \cos(x_{t', j}^{v_1}, x_{t', j}^{v_2}) \quad (4)$$

$x_{t', j}^v$ はタイムステップ t' におけるノード v が保持する j 番目の属性を表しており、主要な属性は p 番目の属性となる。

5.1.2 将来出現するノード数と消失するノード数の予測

将来新たに出現するノード数と消失するノード数の予測を行う。タイムステップ t から L ステップ前までに観測された時系列グラフのノード数の時系列変化を学習し、タイムステップ $t+1$ における未知ノード数、復帰ノード数、消失ノード数をそれぞれ予測する。具体的に用いる手法としては、単純な統計的アプローチの重回帰分析や、時系列分析手法の ARIMA, NN モデルの LSTM が挙げられる。時系列 L の長さや、定常過程および自己相関の有無により適切な手法は異なるため、対象とする時系列データに応じてアプローチを変える必要がある。

5.1.3 未知ノードの属性とリンクの予測

5.1.2 項で予測された未知ノード数のノードを $\mathcal{U}_t(L)$ に追加する。この拡張されたノード集合を $\hat{\mathcal{U}}_{t+1}(L)$ とする。次に、未知ノードが保持する主要な属性を予測する。タイムステップ $t+1$ において、グラフ全体で主要な属性が取り得る値の分布を 5.1.2 節と同様の手法で予測し、これを未知ノードに割り当てる。未知ノード v_* のリンクは、割り当てられた主要な属性に基づいて以下のように予測される。

$$e = \text{NN}([x_{(t+1), p}^{v_*}, x_{i, p}^v]) \in \{0, 1\} \quad (5)$$

NN は入力として、 $x_{(t+1), p}^{v_*}$ と $x_{i, p}^v$ の連結をとり、0 か 1 を出力するニューラルネットワークである。ただし、 $i \in \{t-L+1, t-L+2, \dots, t\}$, $v \in \hat{\mathcal{U}}_{t+1}(L)$, $v \neq v_*$ である。 e が 1 であるときノード v_* とノード v の間にリンク (v_*, v) が存在するとし、これを $\mathcal{D}_t(L)$ に追加する。このリンク集合を $\hat{\mathcal{D}}_{t+1}(L)$ とする。このとき、拡張されたグラフ構造を $\hat{\mathcal{G}}_{t+1}(L) := (\hat{\mathcal{U}}_{t+1}(L), \hat{\mathcal{D}}_{t+1}(L))$ と表記する。 $\hat{\mathcal{G}}_{t+1}(L)$ は特徴抽出アーキテクチャへの入力グラフとなる。

5.1.4 入力グラフのノード属性

$\hat{\mathcal{U}}_{t+1}(L)$ にはタイムステップ t から L ステップ前までに

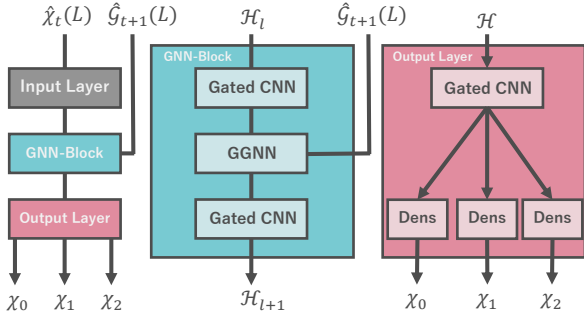


図3 特徴抽出アーキテクチャ概要

観測されたノードと、次のタイムステップ $t+1$ で出現を予測された未知ノードが含まれている。全てのノード $v \in \hat{\mathcal{U}}_{t+1}(L)$ はタイムステップ $t-L+1$ から t までの属性 $\mathcal{X}_t^v(L) := [x_{t-L+1}^v; x_{t-L+2}^v; \dots; x_t^v] \in \mathbb{R}^{L \times n}$ を有しているとする。グラフ $\hat{\mathcal{G}}_{t+1}(L)$ 全体のノード属性を $\hat{\mathcal{X}}_t(L) := \langle \mathcal{X}_t^1(L), \mathcal{X}_t^2(L), \dots, \mathcal{X}_t^{|\hat{\mathcal{U}}_{t+1}(L)|}(L) \rangle \in \mathbb{R}^{|\hat{\mathcal{U}}_{t+1}(L)| \times L \times n}$ と表記する。ただし、 v が存在しないタイムステップ t における属性 x_t^v は NULL であるとする。未知ノードの属性は全てのタイムステップで NULL である。

5.2 特徴抽出アーキテクチャ

この節では、特徴抽出アーキテクチャについての詳細を述べる。図3に特徴抽出アーキテクチャの概要を示す。特徴抽出アーキテクチャは、グラフ $\hat{\mathcal{G}}_{t+1}(L)$ および、 $\hat{\mathcal{X}}_t(L)$ を入力とし、「将来のノードの属性」と「将来復帰するノードの確率」と「将来消失するノードの確率」を出力とする。我々が新たに提案する特徴抽出アーキテクチャは、STGCN では考慮されていない属性付き時系列グラフのトポロジの時間変化に対応できるアーキテクチャである。アーキテクチャは入力層、複数の GNN ブロック、出力層で構成される。時空間グラフ畳込みブロックは2つの Gated CNN 層とその間に1つの GNNs 層を持つサンドイッチ構造として形成される。それぞれの層の詳細は以下の通りである。

5.2.1 入力層

各ノード $v \in \hat{\mathcal{U}}_{t+1}(L)$ は表現ベクトル h_v を有しており、 $h_v = \mathcal{X}_t^v(L) \in \mathbb{R}^{L \times n}$ で初期化を行う。各リンク $e \in \hat{\mathcal{D}}_{t+1}(L)$ には e が存在したタイムステップを表すラベルを付与する。

5.2.2 GNNs 層

GNNs 層ではグラフ全体の空間的依存性を捉える。ノード v について、GNNs 層に入力される表現ベクトルを $h_{vi} \in \mathbb{R}^{M \times C_i}$ とする。 $\hat{\mathcal{U}}_{t+1}(L)$ の各ノード v に対して、同じ GGNN を用いたグラフ畳込み演算が h_{vi} に並列に課される。したがって、グラフ畳込み演算は $\mathcal{H}_i \in \mathbb{R}^{|\hat{\mathcal{U}}_{t+1}(L)| \times M \times C_i}$ として示される3次元テンソルでさらに一般化することができる。表現ベクトルの更新は、式(1)と式(2)の繰り返しによって行われ、 $\mathcal{H}_o = \text{GGNN}(\mathcal{H}_i) \in \mathbb{R}^{|\hat{\mathcal{U}}_{t+1}(L)| \times M \times C_o}$ が出力される。

5.2.3 Gated CNN 層

Gated CNN 層では属性の時間的依存性を捉える。ノード v について、入力される表現ベクトルを $h_{vi} \in \mathbb{R}^{M \times C_i}$ とする。時間畳込み演算は式(3)によって行われ、 $h_{vo} \in \mathbb{R}^{(M-K_t+1) \times C_o}$

が出力される。GNNs 層と同様に、Gated CNN 層もまた、全てのノードに同じ畳込みカーネル Γ を採用することによって、 $\mathcal{H}_i \in \mathbb{R}^{|\hat{\mathcal{U}}_{t+1}(L)| \times M \times C_i}$ として示される3次元テンソルで一般化することができる。このとき、 $\mathcal{H}_o = \Gamma * (\mathcal{H}_i) \in \mathbb{R}^{|\hat{\mathcal{U}}_{t+1}(L)| \times (M-K_t+1) \times C_o}$ 。

5.2.4 GNN ブロック

GNN ブロックは1層の GNNs 層を2層の Gated CNN 層で挟むことで、トポロジ変化と属性変化を同時に学習する。ブロックは対象とするケースの規模と複雑さに応じて積み重ね、拡張することができる。また、ボトルネック戦略として、GNNs 層を通じたチャンネル C の縮小と拡大によって、スケール圧縮と特徴スキージング [18] が適用される。過学習を防ぐため全てのブロック内で正規化を行う。GNN ブロックの入出力はすべて3次元テンソルとなる。ブロック l の入力 $\mathcal{H}_l \in \mathbb{R}^{|\hat{\mathcal{U}}_{t+1}(L)| \times M \times C_i}$ に対して、出力 $\mathcal{H}_{l+1} \in \mathbb{R}^{|\hat{\mathcal{U}}_{t+1}(L)| \times (M-2(K_t-1)) \times C_o}$ は以下のように計算される。

$$\mathcal{H}_{l+1} = \Gamma_1^l * \text{GGNN}(\Gamma_0^l * \mathcal{H}_l) \quad (6)$$

Γ_0^l および Γ_1^l はそれぞれブロック l 内の Gated CNN 層における畳込みカーネルである。

5.2.5 出力層

最後に出力層として、マルチタスク学習を行うための複数の全結合層に接続される追加の Gated CNN 層を接続する。この Gated CNN 層は最後の GNN ブロックの出力を単一ステップの表現ベクトル $Z \in \mathbb{R}^{|\hat{\mathcal{U}}_{t+1}(L)| \times C}$ に写像する。特徴抽出アーキテクチャの出力要件として(1) 将来のノードの属性、(2) 将来復帰するノードの確率、(3) 将来消失するノードの確率の出力が求められるが、(1) は、 C 次元から n 次元に写像する線形変換 $\mathcal{X}_0 = Z w_0 + b_0 \in \mathbb{R}^{|\hat{\mathcal{U}}_{t+1}(L)| \times n}$ を適用することで全ノードの将来の属性が計算される。ここで $w_0 \in \mathbb{R}^{C \times n}$ は重み行列、 b_0 はバイアスである。(2) については $\mathcal{X}_1 = \sigma(Z w_1 + b_1) \in \mathbb{R}^{|\hat{\mathcal{U}}_{t+1}(L)|}$ を適用し、全てのノードについて復帰する確率を計算する。ここで $w_1 \in \mathbb{R}^C$ は重み行列、 b_1 はバイアスである。(3) については $\mathcal{X}_2 = \sigma(Z w_2 + b_2) \in \mathbb{R}^{|\hat{\mathcal{U}}_{t+1}(L)|}$ を適用し、全てのノードについて消失する確率を計算する。ここで $w_2 \in \mathbb{R}^C$ は重み行列、 b_2 はバイアスである。

損失関数として、連続量変数の予測では mean squared error (loss_c) を、カテゴリカル変数の予測では softmax cross entropy (loss_d) を用いる。復帰ノードおよび、消失ノードの予測の損失関数に binary cross entropy を用いる。したがって、特徴抽出アーキテクチャの損失関数は以下の通りとなる。

$$\begin{aligned} L(\mathcal{X}_0, \mathcal{X}_1, \mathcal{X}_2) = & \text{loss}_c(\mathcal{X}_0, X_{t+1}) \\ & + \text{loss}_d(\mathcal{X}_0, X_{t+1}) \\ & + \sum_v -(T_{r_v} \mathcal{X}_{1v} + T_{l_v} \mathcal{X}_{2v}) \end{aligned} \quad (7)$$

$T_r \in \mathbb{R}^{|\hat{\mathcal{U}}_{t+1}(L)|}$ は復帰するノードを表す要素は1、それ以外の要素は0を有している教師ベクトルである。 $T_l \in \mathbb{R}^{|\hat{\mathcal{U}}_{t+1}(L)|}$ は消失するノードを表す要素は1、それ以外の要素は0を有し

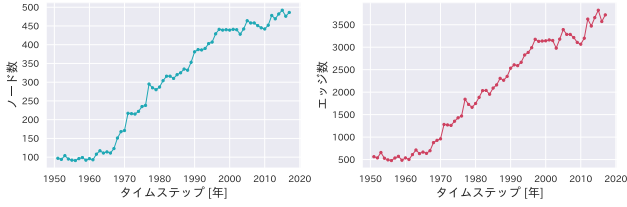


図4 ノード数の時系列変化 図5 エッジ数の時系列変化

表1 NBA ソーシャルネットワークの統計量

	平均	標準偏差	最大値	最小値
ノード数	298.05	142.99	492.0	91.0
エッジ数	2019.6	1115.2	3823.0	479.0

ている教師ベクトルである。

5.3 グラフ生成

$\bar{V}_{t+1} = \emptyset$, $\bar{E}_{t+1} = \emptyset$, $\bar{X}_{t+1} = \emptyset$ とする。まず、タイムステップ t におけるノード集合 \mathcal{V}_t を \bar{V}_{t+1} に、リンク集合 \mathcal{E}_t を \bar{E}_{t+1} に追加する。次に、消失する確率の高いノードから順に 5.1.2 項で予測された消失ノード数だけ \bar{V}_{t+1} から削除し、これに接続されているリンクを \bar{E}_{t+1} から削除する。同様に、復帰する確率の高いノードから順に予測された復帰ノード数だけ \bar{V}_{t+1} に追加し、これと接続可能なリンクを \bar{E}_{t+1} に追加する。その後、未知ノードとこれに接続されるリンクをそれぞれ追加する。最後に、ノード $v \in \bar{V}_{t+1}$ が保持する属性 \mathcal{X}_{0v} を \bar{X}_{t+1} に追加する。タイムステップ $t+1$ の属性付き時系列グラフの予測結果を $\bar{G}_{t+1} = (\bar{V}_{t+1}, \bar{E}_{t+1}, \bar{X}_{t+1})$ とする。

6. 評価実験

本章では提案フレームワークの予測精度を評価する。

6.1 データセット

1951年から2017年までのNBAの67シーズンにおける選手の統計値、ボックススコアを用いて、ソーシャルネットワークを構築し、提案フレームワークの実験と評価を行う。データセットはBasketball Reference^(注1)から引用している。NBAソーシャルネットワークは選手をノード、同じチームの選手に対してリンクを張ることで属性付き時系列グラフとして表現できる。ノードの属性として、所属チーム、ポジション、年齢、得点を有しているものとする。また、シーズン内における選手のチーム移籍はないものとし、2017年まででdefunctなチームに所属していた選手は除外している。表1、図4、および図5にNBAソーシャルネットワークの統計量を示す。

6.2 実験設定

NBAデータセットからは、67ステップの属性付き時系列グラフが構築される。過去 L ステップのグラフを用いて将来予測を行うとき、得られるサンプル数は $67-L$ となる。例えば、 $L=3$ の時、1951年、1952年、1953年のグラフから1954年のグラフを予測するので、1つのサンプルは $\{\mathcal{G}_{1951}, \mathcal{G}_{1952}, \mathcal{G}_{1953}, \mathcal{G}_{1954}\}$ となる。本稿では $L=3$ とし、評価用として10個のサンプル、残りのサンプルを学習用に用いる。評価は、提案フ

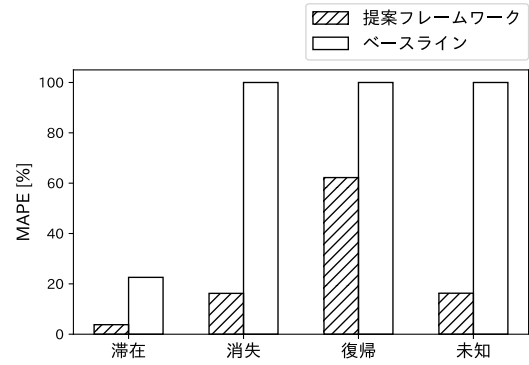


図6 ノード数の予測誤差率

表2 ノードの特定精度

手法 (ノードの種類)	再現率	適合率	F 値
提案フレームワーク (滞在)	0.8932	0.8977	0.8949
提案フレームワーク (消失)	0.5469	0.5342	0.5355
提案フレームワーク (復帰)	0.3138	0.3319	0.299
ベースライン (滞在)	1.0000	0.8160	0.8985
ベースライン (消失)	0.0000	-	-
ベースライン (復帰)	0.0000	-	-

レームワークが出力したグラフ \bar{G}_{t+1} を予測結果とする場合と、ベースラインとしてタイムステップ t のグラフ \mathcal{G}_t をそのまま出力する場合の比較を行う。

6.3 実験結果

提案フレームワークのノード予測の精度、属性予測の精度、リンク予測の精度について評価を行う。

6.3.1 ノード予測

ノードの種類ごとに評価を行う。まず、予測結果として出力されたグラフのノード数の予測誤差を平均絶対誤差率 (MAPE) で評価する。図6において、左から順に滞在ノード数の誤差率、消失ノード数の誤差率、復帰ノード数の誤差率、未知ノード数の誤差率を表している。ベースラインは全てのノードを滞在ノードと予測しているのに対し、提案フレームワークは滞在ノードだけでなく、消失ノード、復帰ノード、未知ノードの予測が可能であるため、全ての種類のノード数の予測誤差率はベースラインより大幅に小さくなっている。

次に、ノードの特定精度について再現率、適合率および、F値で評価を行う (表2)。予測された未知ノードは真の未知ノードとの一致判定が不可能であるため特定精度の評価から除外している。ベースラインでは前タイムステップのグラフと同じグラフを予測結果としているため、滞在ノードの再現率は1となるが、消失、あるいは復帰するノードを予測することができないため、滞在ノードの適合率は著しく下がる。提案フレームワークでは、消失ノードを54%、復帰ノードを33%特定できている。一方で、滞在ノードを消失ノードと誤判定する可能性があるため、滞在ノードの再現率が下がっているが、適合率ではベースラインを大幅に上回る結果となっている。

6.3.2 属性予測

ノードは属性として年齢、得点、所属チーム、ポジションを保持しており、それぞれの属性の予測結果について評価を行う。

(注1) : <https://www.basketball-reference.com/>

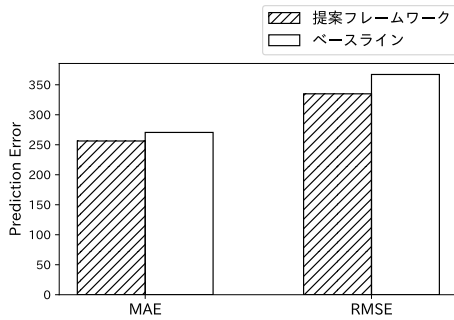


図7 属性の予測結果 (得点)

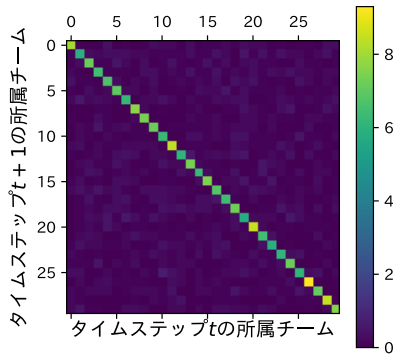


図8 チームの移籍先ヒートマップ

年齢については予測の必要が無いため省略する。未知ノードの属性は真の未知ノードとの一致判定が不可能であるため、まず、一致判定が可能なノードのみについて属性予測の精度評価を行う。

(1) 得点の予測 : 選手が次のシーズンで獲得する得点数を予測する。回帰問題であるため平均絶対誤差 (MAE) と根平均二乗誤差 (RMSE) で評価を行う。図7において、MAE と RMSE はどちらも誤差の指標で、小さい方がより良い精度で予測できていることを表す。また、RMSE は MAE よりも外れ値をより大きく扱う特徴を有する。ベースラインは学習系列の最終時点の値を出力するナイーブな予測手法であるが、提案フレームワークは MAE と RMSE の双方において、これより高精度で予測可能であることを示している。このことから、提案フレームワークが、属性の時間変化をモデル化できていることが分かる。

(2) 所属チームとポジションの予測 : NBA は 30 チーム、11 のポジションが存在する。どちらも多クラス分類問題に帰着できるので、正答率で評価を行う (表3)。まず、所属チームについて評価する。ベースラインの正答率が 62% であるのに対し、提案フレームワークの正答率は 59% と低くなっている。これについて図8を用いて説明する。図8は選手のチームの移籍先をヒートマップで可視化したものである。軸の目盛りはチームのインデックスに対応している。例えば、0行5列目の要素はチーム5からチーム0へ移籍した選手の平均人数を表している。対角成分は次のシーズンで移籍せずと同じチームで試合に出場する選手の平均人数を表す。対角成分の他に色が濃い要素はなく、チームの移籍先について規則性は確認できない。このことから、提案フレームワークは所属チームについては、タイ

表3 所属チームとポジションの予測精度

属性	正答率
所属チーム (提案フレームワーク)	0.5907
所属チーム (ベースライン)	0.6235
ポジション (提案フレームワーク)	0.7985
ポジション (ベースライン)	0.8163

表4 リンク予測の精度

手法 (リンクの種類)	再現率	適合率	F 値
提案フレームワーク (滞在)	0.3193	0.3050	0.3115
提案フレームワーク (消失)	0.5291	0.5904	0.5545
提案フレームワーク (復帰)	0.0428	0.0730	0.0489
ベースライン (滞在)	0.3686	0.2438	0.2933
ベースライン (消失)	0.0000	-	-
ベースライン (復帰)	0.0000	-	-

ムステップ t の所属チームをそのまま予測結果として出力するよう正しく学習が行われたと考えられる。ベースラインがタイムステップ t と同じチームを予測結果とするのに対し、提案フレームワークは他のチームを予測する可能性を含んでいるため、正答率はやや下がる結果となっている。ポジションについても同様の理由により、提案フレームワークが属性変化の学習に失敗しているわけではないことを説明できる。

(3) 未知ノードの主要な属性の予測精度 : 未知ノードの属性について、主要な属性 (所属チーム) ごとの出現頻度の分布を \cos 類似度で評価する。図9に真の未知ノードのヒストグラム平均と予測された未知ノードのヒストグラム平均を示す。 \cos 類似度は 0.8174 となっており、未知ノードの属性についておおよそ予測できていることがわかる。ベースラインについては未知ノードの出現を予測していないため省略する。

6.3.3 リンク予測

リンクの種類ごとに評価を行う。まず、予測結果として出力されたグラフのリンク数の予測誤差を MAPE で評価する。図10において左から順に、滞在ノード、消失ノード、復帰ノード、未知ノードに接続されるリンク数の予測誤差率を表す。ベースラインは全てのリンクを滞在ノードに接続されるリンクと予測するのに対し、提案フレームワークは滞在、消失、復帰、未知それぞれについて予測可能であるため、予測誤差率は小さくなっている。次に、リンク予測の精度について再現率、適合率および、F 値で評価を行う (表4)。提案フレームワークは消失するリンクも予測可能であり、滞在ノードに接続されるリンクを消失するリンクと誤判定する可能性があることから、滞在ノードに接続されるリンクの再現率はベースラインのものより低くなっているが、適合率および、F 値においてはベースラインを上回る結果となっている。

7. 結論

本稿では、未知ノードの出現を考慮した属性付き時系列グラフ予測のための新しい深層学習フレームワークを提案した。特徴抽出アーキテクチャでは GNNs 層と Gated CNN 層を GNN ブロックを通して統合し、ノード数の時間変化に対応できるよ

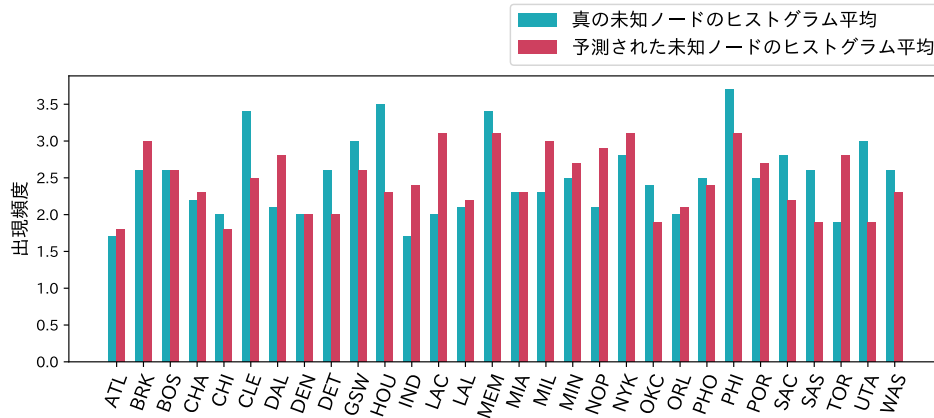


図9 未知ノードの属性評価

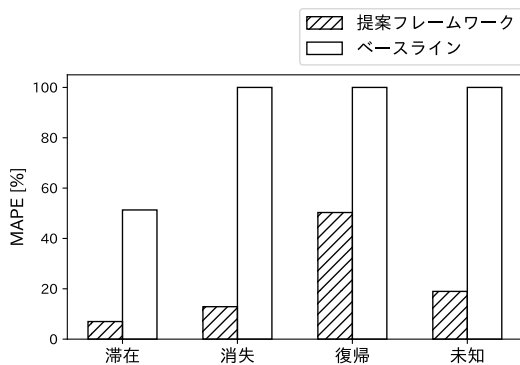


図10 リンク数の予測誤差率

うにした。実験では、我々のモデルが実世界のデータセットに対して既存の研究では取り組むことの出来なかった問題に取り組めることを示し、入力グラフの属性変化とトポロジ変化を同時に考慮した予測を実現した。

謝 辞

本研究は科学研究費（JP16H01722）の支援によって行われた。ここに記して謝意を表す。

文 献

- [1] Yu Bing, Haoteng Yin, and Zhu Zhanxing. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In *Proceedings of the IJCAI*, 2018.
- [2] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *Proceedings of the ICLR*, 2018.
- [3] Huiyuan Chen and Jing Li. Exploiting structural and temporal evolution in dynamic link prediction. In *Proceedings of the ACM CIKM*, pp. 427–436, 2018.
- [4] Yann N Dauphin, Angela Fan, Michael Auli, and David Grangier. Language modeling with gated convolutional networks. *arXiv preprint arXiv:1612.08083*, 2016.
- [5] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. Gated graph sequence neural networks. In *Proceedings of the ICLR*, 2016.
- [6] Jia Yuhan, Wu Jianping, and Du Yiman. In *Proceedings of the IEEE ITSC*, pp. 1217–1222, 2016.
- [7] Yisheng Lv, Yanjie Duan, Wenwen Kang, Zhengxi Li, Fei-Yue Wang, et al. Traffic flow prediction with big data: A deep learning approach. *The IEEE TITS*, Vol. 16, No. 2, pp. 865–873, 2015.
- [8] Quanjun Chen, Xuan Song, Harutoshi Yamada, and Ryosuke Shibasaki. Learning deep representation from big and heterogeneous data for traffic accident inference. In *Proceedings of the AAAI*, pp. 338–344, 2016.
- [9] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *The MIT Press Neural Computation*, Vol. 9, No. 8, pp. 1735–1780, 1997.
- [10] Yuankai Wu and Huachun Tan. Short-term traffic flow forecasting with spatial-temporal correlation in a hybrid deep learning framework. *arXiv preprint arXiv:1612.01022*, 2016.
- [11] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*, 2013.
- [12] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Proceedings of the NIPS*, pp. 3844–3852, 2016.
- [13] David Liben-Nowell and Jon Kleinberg. The link-prediction problem for social networks. *The JASIST*, Vol. 58, No. 7, pp. 1019–1031, 2007.
- [14] Evrim Acar, Daniel M Dunlavy, and Tamara G Kolda. Link prediction on evolving data using matrix and tensor factorizations. In *Proceedings of the IEEE ICDM Data Mining Workshops*, pp. 262–269, 2009.
- [15] Daniel M. Dunlavy, Tamara G. Kolda, and Evrim Acar. Temporal link prediction using matrix and tensor factorizations. *The ACM*, pp. 1–27, 2011.
- [16] Jundong Li, Kewei Cheng, Liang Wu, and Huan Liu. Streaming link prediction on dynamic attributed networks. In *Proceedings of the ACM WSDM*, pp. 369–377, 2018.
- [17] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.
- [18] Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. *arXiv preprint arXiv:1704.01155*, 2017.