DEIM2021 H21-2

グラフニューラルネットワークを用いたスプレッドシートの見出し認識

笹治　拓矢†　　加藤　　誠††

† 筑波大学 知識情報・図書館学類　〒 305–8550 茨城県つくば市春日 1-2
†† 筑波大学 図書館情報メディア系　〒 305–8550 茨城県つくば市春日 1-2
E-mail: †s1913574@s.tsukuba.ac.jp, ††mpkato@acm.org

あらまし　本論文では，スプレッドシート内の統計表を構成するセルの視覚的な情報（文字列や値，罫線など）をグラフデータに変換し，グラフニューラルネットワーク (GNN) を適用することでスプレッドシートの大域的な特徴を考慮して統計表の見出し認識を行う方法を提案する．また，教師データを用意するためには大きな労力が必要となるため，教師なし表現学習を行うことで有効なセル表現を学習し，少数の教師データであっても効果的な学習が可能となる方法を提案する．実験では，政府統計ポータルサイト e-Stat で公開されているスプレットシートを使用し，GNN の教師あり学習手法が従来の機械学習による手法よりも見出し階層の認識で高い精度を達成できることを示した．

キーワード　スプレッドシート，グラフニューラルネットワーク，教師なし学習，統計表の意味解釈，見出し認識

1　はじめに

政府や企業により，膨大な各種データが統計表データとしてウェブ上で広く公開されている．そのため，多くの統計データを整理・分析することで知見を得て，仕事などに役立てることができる．ただし，これらのデータは Excel 形式や CSV 形式による表形式データとして公開されており，中には見出しを階層関係にするなど複雑な階層関係を含むものも存在する．

大量の統計データから知見を得るには，機械的な処理が安易となるデータに変換する必要があるため複雑な階層関係の認識タスクのほかに様々な正規化に関するタスクが提案されてきた．特にスプレッドシートの正規化は，機械的な処理が困難であるシート内の統計表に対して，見出し階層の認識や表構造の意味を解釈する手法を適用することで，リレーショナルデータとして抽出できる形に変換するタスクである．具体的なタスクとして，シート内のどの部分に統計表を記述されているかを認識する表認識タスクがある．その他には，統計表における役割（見出しやデータ，タイトルなど）を識別するセル種別の識別タスクがあり，既存研究ではセルベクトル表現を用いた RNN ベースの分類手法が提案されており，セル種別の分類精度を向上するために文体的特徴と組み合わせている [1]．見出し階層の認識タスクの既存研究では，複雑な構造をもつ統計表を画像化し，表の局所的な特徴（罫線の有無や値間の位置関係）に基づいて階層関係を判定している [2]．しかし，局所的な特徴による認識では，複数の罫線デザインで記述した見出しや，罫線そのものを使用せずに，図形を挿入した形で見出し階層を表現した場合に，分類器が誤認識してしまう問題がある．

本論文では，スプレッドシート内の統計表を構成するセルの視覚的な情報（文字列や値，罫線など）をノード特徴とするグラフデータに変換し，グラフニューラルネットワーク (GNN) を適用することで，スプレッドシートの大域的な特徴を考慮して統計表における見出しとその階層関係の認識（見出し認識）を行う方法を提案する．また，見出し認識の教師データを用意するためには大きな労力が必要となるため教師なし表現学習を行うことによって認識タスクに有効なセル表現を学習し，少数の教師データであっても効果的な学習が可能となる方法を提案する．具体的には，教師なし表現学習手法の Deep Graph Infomax (DGI) [3] で認識タスクに有効な特徴表現を獲得する．DGI の事前学習は，相互情報量の最大化を学習タスクとして，大量の正解のないデータを活用する．

本タスクに有効な特徴表現を得るために DGI の目的関数をグラフ上の局所領域の特徴（パッチ表現）と大域的な特徴の間の相互情報量の最大化とすることで，統計表全体に関連のある特徴表現を学習過程で得られるようにすることができる．GNN の教師なし表現学習手法の中で DGI の学習表現はノード分類タスクに適用でき，また，得られる特徴表現がインダクティブな分類タスク（訓練とテストで，異なるグラフデータが与えられる設定）で一貫して性能が高いことから本手法を採用した．我々の知る限りにおいて，本研究は見出し階層の認識タスクに GNN の教師なし学習手法を直接活用する最初の試みである．従来の手法では，セル種別を識別した後に見出し階層の認識を行っており，事前学習で有効な特徴表現を獲得する方法を見出し階層の認識タスクに応用されていない．

実験では，見出し階層のセルペアの認識と見出しセルの識別を対象とし，比較手法を従来の機械学習による手法として，事前学習済みの DGI から得られる埋め込みを用いて分類器の評価を行った．データセットには，政府統計ポータルサイト e-Stat[1] から収集されたスプレットシートを使用し，認識タスクの正解ラベルを用意するためにアノテーションを行った．学習で使用

1：https://www.e-stat.go.jp/

する特徴量は，スプレッドシート内におけるセル情報を中心とした 11 種類で構成されている．例えば，入力値（セル値）や文字サイズ，セル結合の有無である．実験の結果，GNN の教師あり学習手法が従来の機械学習による手法よりも見出し階層の認識で高い精度を達成できることを示した．また，DGI の埋め込みを用いた分類器の精度は GNN の教師あり学習手法よりも低くなった．見出しセルの識別ではノード初期特徴ベクトルを用いた分類器の精度が最も高くなることを示した．このことから，GNN の教師あり学習手法は見出し階層の認識に効果的であることが明らかになった．

この論文における我々の貢献を以下に示す：(1) スプレッドシート内の統計表を構成するセルの視覚的な情報（文字列や値，罫線など）を特徴とした，GNN による教師なし学習を適用した．(2) スプレッドシートに対してアノテーションを行い，見出しとその階層関係の認識タスクのためのデータセットを構築した．(3)GNN の教師あり学習手法が見出し階層の認識タスクにおいて効果的であることを，実験によって明らかにした．

本論文の構成は以下の通りである．2 節ではスプレッドシートの正規化と GNN に関する関連研究ついて述べる．3 節では問題設定を説明し，GNN による手法，および，その主問題への適用方法について述べる．4 節では実験結果を示す．最後に，5 節では今後の課題と共に本論文の結論を述べる．

2 関 連 研 究

本節では，スプレッドシートの（統計表の）正規化に関する関連研究について述べる．

表構造の意味解釈に関する研究では，表形式セルの様式的特徴（文字サイズ，罫線種類，背景色など）を用いて，統計表におけるセル種別（見出し，データなど）を分類する手法が提案されている [2], [4], [5], [6].

松井らは，ある対象の時間的変化を記述した動向情報を対象に，その客観的根拠を統計表データから探索するシステムを構築するために，クエリの根拠となる統計表データを理解しやすい形に変換する目的で表構造を認識する手法を提案している [4]．手法自体は，ルールベースであることから，見出しの範囲がタイトルや注釈のセルまで含んでしまう誤認識が発生し，シート内の複数の統計表にも対応できない．この研究では，見出しにおける階層関係の認識も取り組んでいるが適切な評価がされておらず，階層認識の課題も明確に示されていない．同著者らは，統計表の構造認識を系列ラベリング問題として解釈し，条件付き確率場 (CRF) を用いた識別モデルで実現する手法も提案している [5]．結果として，統計表におけるタイトルや注釈などのセル範囲を認識することができたが，見出し階層を含んだ見出しの範囲を認識するのが困難であった．また，2 次元の表データを 1 次元に変換してから学習するアプローチを採用したため，2 次元の形式を活かした学習手法とはなっていない．曽和らは，複雑な構造をもつ統計表を，一旦画像化して解析することにより，表の見た目に応じた柔軟な見出し階層の認識と，表構造の意味解釈を実現する手法を提案している [2]．しかし，セル内容は，OCR による文字認識を採用したことから，見出しに含まれる日本語や英語の認識精度が低くなっている．また，この手法は表の局所的な特徴（罫線の有無や値間の位置関係) に基づいて階層関係を判定しているため，複雑な見出し階層では誤認識が起きてしまう．Koci らも，表領域を認識するためにルールベースの手法を提案している [6]．手法としては，シート内の個々のセル種別を推定した後，それらの領域を構築している．この手法の利点は，シート内の表が単一ではなく任意の数を認識でき，分類ミスや空白セルなどの不規則性があっても認識できる点である．しかし，複数の表領域を認識するために，見出し階層を考慮しておらず，同じセル種別の領域として処理している．

また，事前学習で得られる表形式セルの埋め込みを分類モデルに学習させる手法も提案されている．Ghasemi らは，シート内のセルに対して局所的な文脈情報を取り込む文脈的セル埋め込みとスタイル情報の文体的セル埋め込みの両方を用いて，シート内の統計表におけるセル種別を分類している [1].

スプレッドシート内の表のセル種別を識別するタスクにおいて，手法の精度を評価するために用いられる既存のデータセットとして，DeEx [7] と SAUS [1] および CIUS [1] がある．DeEx は，DeExcelerator プロジェクト [2]で作成されたデータセットとなっており，ENRON [8]，FUSE [9]，EUSES [10] という 3 つのスプレッドシートのコーパスで構成されている．SAUS と CIUS は，Ghasemi らがスプレッドシート内の表のセル種別を手作業でラベル付けして作成されたものである [1].

3 提 案 手 法

本節では，スプレッドシート上におけるセル内容とスタイル情報から，見出しとその階層関係を学習する問題についての説明を行う．さらに，グラフニューラルネットワークの手法を導入し，その問題への適用について述べる．

3.1 問 題 設 定

スプレッドシート s は n_s 行，m_s 列のセルで構成されており，i 行，j 列目のセルを c_{ij} とする．このとき，見出しセルの識別タスクは，スプレッドシート s 内のセル c_{ij} が与えられたときに，そのセルが見出しであるかどうかを判定する問題と定義する．また，見出し階層関係の認識タスクは，スプレッドシート s 内のセル c_{ij}，c_{kl} が与えられたときに，それらが階層関係にあるかどうかを判定する問題と定義する．ただし，本タスクの学習で使われる特徴量には，スプレッドシート s 内のセル c_{ij} に関する情報，具体的にはセル値（入力された値）とスタイル情報（セルの幅や高さなど）を組み合わせた特徴を表す F 次元のベクトルを考えるとする．次節では，ここまでのスプレッドシートを学習単位とした問題設定を本手法で扱えるグラフデータとして解釈する．

2：https://wwwdb.inf.tu-dresden.de/research-projects/deexcelarator/

3.2 提案手法の概要

我々の主目的は，スプレッドシート s 上の統計表を構成するセルの内容とスタイル情報を各セル c_{ij} の特徴 F 次元ベクトルとしたシートデータから，本タスクに有効な特徴表現（セル埋め込み）F' 次元ベクトルを教師なし表現学習手法で学習することである．

次の段階では，教師なし表現学習手法で得られたセル埋め込み F' 次元ベクトルを分類器に用いることで見出しとその階層関係の認識を行う．分類器は，用意した正解クラス情報とその特徴ベクトル群を用いて学習し，タスクに応じた特徴ベクトルを所定のクラスへ分類し，クラス情報を出力する．この際，見出し階層の認識タスクの学習には，任意のセル対の埋め込みを連結したベクトル $[c_{ij}; c_{kl}]$ 群と正解のクラス情報を用いる．また，ベクトルの連結を $[\ldots;\ldots]$ と表す．

本研究では，見出しとその階層の認識タスクの学習の際にスプレッドシートを変換したグラフデータを活用する．また，グラフデータは，表構造の意味解釈する上で各セルの隣接関係やセル自体の特徴を持たせることが可能である．前節を踏まえ，以下を定義する．スプレッドシートをグラフ $G = (V, E)$ として，V はノード（セル）の集合，E はエッジの集合（セルの隣接関係）$E \subseteq V \times V$ を表すこととする．また，ノードを $v_i \in V$，エッジを $e_{ij} = (v_i, v_j) \in E$ と表し，隣接するノードの集合を $\mathcal{N}(v) = \{u \in V | (v, u) \in E\}$ とする．

上記を踏まえ，本タスクの問題設定をグラフを用いて再定義する．見出しセルの識別タスクは，グラフ $G = (V, E)$ が与えられたときに，各ノード $v \in V$ が見出しであるかどうかを判定する問題となる．また，見出し階層関係の認識タスクはグラフ G が与えられたときに，階層関係にあるすべてのノード対 $P \subseteq V \times V$ を求める問題となる．

さらに，本研究ではグラフに関して以下を定義する．グラフ G のノード数を $N = |V|$ とすることで，ノード集合を $V = \{v_1, \ldots, v_N\}$ として，あるノード間 (v_i, v_j) にエッジがあるかどうか，言い換えると，2 つのノードが隣接しているかを隣接行列 $\mathbf{A}_{ij} \in \mathbb{R}^{N \times N}$ で表す．すなわち，グラフ内の v_i と v_j にエッジが存在する場合は $\mathbf{A}_{ij} = 1$，そうでない場合は $\mathbf{A}_{ij} = 0$ とする．各ノードの次数（接続されたエッジ数）を対角成分に並べた行列を次数行列 $\mathbf{D}$ とし，グラフ内の v_i の次数は $\mathbf{D}_{ii} = \sum_k \mathbf{A}_{ik}$ となる．全ノードの初期特徴ベクトルを $\mathbf{X} \in \mathbb{R}^{N \times F}$ として，各ノード v_i の初期特徴ベクトルを $\mathbf{x}_i \in \mathbb{R}^F$ と表現する．また，本タスクの分類問題を解くために，各ノード $v \in V$ に見出しのラベル，ノード対 P に見出し階層のラベルを割り当てることにする．

本研究では，見出し階層を認識する問題をグラフデータのリンク予測問題としても解釈できることを示す．ここでいうリンク予測問題は,「グラフ構造の既知の部分が与えられたときに，未知の部分を予測する問題」と定義する．具体的には，隣接ノードにエッジを張ることで構成されたスプレッドシートグラフの一部分を手がかりに，まだ知られていない階層関係を予測する問題を考えるとすると，部分的に分かっているグラフ構造を手がかりにそれ以外の部分を予測するという「グラフの補完問題」と捉えることができる．グラフ構造の既知の部分に関しては， 正例と負例が分かる場合と，リンクがあると分かっている場所だけが与えられる場合, つまり, 正例のみで学習を行う場合の 2 種類があるが，本研究は前者として問題を扱う．また，リンクの有無は独立に予測するとして，ノードペアの 2 値分類問題としてタスクを解く．つまり，2 つのノード特徴ベクトルに基づいて， 階層関係がもつ性質についての予測を行っていることになる．ノードペア (v_i, v_j) に対応する特徴ベクトル $\mathbf{z}_{ij}$ は，教師なし表現学習手法で得られたノード埋め込み $\mathbf{f}_i$ 及び $\mathbf{f}_j$ の連結 $[\mathbf{f}_i; \mathbf{f}_j]$ として定義する．新しく作られた特徴ベクトルの次元数は，ノード埋め込みを F' 次元とすると $2F'$ 次元となる．リンク予測は，例えば全結合層のネットワークを用いてリンクの度合いを出力する関数 h によって入力特徴ベクトル $\mathbf{z}_{ij}$ を 2 次元ベクトル $\mathbf{y} \in \mathbb{R}^2$ に変換し，その出力層の活性化関数をソフトマックス関数とすることでリンクの有無 $\hat{t} \in \{0, 1\}$ を出力する．

$$\mathbf{y} = h\left(\mathbf{z}_{ij}\right) \tag{1}$$

$$\hat{t} = \mathrm{argmax}_t \frac{\exp\left(\mathbf{y}_t\right)}{\sum_k \exp\left(\mathbf{y}_k\right)} \tag{2}$$

関数 h の入力には，ノードペア (v_i, v_j) の埋め込みを受け取る．見出し階層の有無は $\hat{t} \in \{0, 1\}$ の 2 値で表される．また，見出しセルの識別タスクも同様の方法で各ノードが見出しであるかを判定する．その際，入力には各ノードの埋め込み $\mathbf{f}_i$ を用いる．

3.3 グラフニューラルネットワークによる手法

グラフニューラルネットワークの代表的な手法として，Graph Convolutional Network (GCN) [11] と Graph Attention Network (GAT) [12] について述べる．

GCN は, グラフ $G = (V, E)$ において，隣接行列 $\mathbf{A} \in \mathbb{R}^{N \times N}$ と特徴行列 $\mathbf{X} \in \mathbb{R}^{N \times F}$ を入力とする非線形関数 f として定義できる．ここでは，ノードの初期特徴ベクトルを F 次元，第 l 層の潜在特徴行列 $\mathbf{H}^{(l)}$ のノード潜在ベクトルを $\mathbf{h}_i^{(l)} \in \mathbb{R}^{F^{(l)}}$ とする．一般的なグラフニューラルネットワークの非線形関数は以下のように定義する．

$$\mathbf{H}^{(l+1)} = f\left(\mathbf{H}^{(l)}, \mathbf{A}\right) \tag{3}$$

ただし，$\mathbf{H}^{(0)} = \mathbf{X}$，$\mathbf{H}^{(L)} = \mathbf{Z} \in \mathbb{R}^{N \times F^{(L)}}$ として，N はグラフのノード数，L は層数とする．関数 f は，重み行列 $\mathbf{W}$ と ReLU などの非線形な活性化関数 $\sigma(\cdot)$ による演算を行う．

$$f(\mathbf{H}^{(l)}, \mathbf{A}) = \sigma\left(\mathbf{A}\mathbf{H}^{(l)}\mathbf{W}^{(l)}\right) \tag{4}$$

しかし，上記の演算では，$\mathbf{A}$ の行列積はすべての隣接ノードの特徴ベクトルを抽出するが，自身のノードの特徴ベクトルを抽出できず，かつ，$\mathbf{A}$ が正規化されていないため，行列積により特徴ベクトルのスケールを変えてしまうという 2 つの問題がある．その問題を解決するために，Kipf ら [11] は以下の 2 つの対策を行っている．

（1） $\mathbf{A}$ に単位行列 $\mathbf{I}_n \in \mathbb{R}^{N\times N}$ を足した $\hat{\mathbf{A}} = \mathbf{A} + \mathbf{I}_n$ を隣接行列として用いる．

（2） $\mathbf{D} \in \mathbb{R}^{n\times n}$($\mathbf{A}$ の次数行列) を用いて，$\mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$ とした正規化ラプラシアン行列に変換する．

2 つの対策によって，GCN の第 l 層における演算は以下のように表される．

$$f(\mathbf{H}^{(l)}, \mathbf{A}) = \sigma\left(\hat{\mathbf{D}}^{-1/2}\hat{\mathbf{A}}\hat{\mathbf{D}}^{-1/2}\mathbf{H}^{(l)}\mathbf{W}^{(l)}\right) \tag{5}$$

ただし，$\hat{\mathbf{D}}$ は $\hat{\mathbf{A}}$ の次数行列，$\mathbf{W}^{(l)}$ は第 l 層の学習可能な重み行列を表すものとする．そして，ノード v_i における畳み込み演算は以下の式で表される．

$$\mathbf{h}_i^{(l+1)} = \sigma\left(\mathbf{W}_0^{(l)\top}\mathbf{h}_i^{(l)} + \sum_{j\in\mathcal{N}_i} c_{ij}\mathbf{W}_1^{(l)\top}\mathbf{h}_j^{(l)}\right) \tag{6}$$

ただし，$c_{ij} = 1/\sqrt{\mathbf{D}_{ii}\mathbf{D}_{jj}}$ は正規化のための定数，$\mathcal{N}_i$ は隣接ノードの集合，$\mathbf{W}_0^{(l)}, \mathbf{W}_1^{(l)}$ は第 l 層の学習可能な重み行列とする．この演算により，隣接ノードの情報を集約して足し合わせた後，得られた情報を用いて自身の特徴表現を更新することができる．

GAT は，隣接ノード $\mathcal{N}(v)$ からの伝搬時に重要度を計算する仕組みを導入した GCN である．各ノードの特徴ベクトル $\mathbf{h}_i \in \mathbb{R}^F$ は隣接ノードの情報を取り込んで更新することで，潜在特徴ベクトル $\mathbf{h}'_i \in \mathbb{R}^{F'}$ が得られる．ただし，N はノード数で，F は初期特徴ベクトルの次元数とする．

一般的なアテンション係数 $e_{ij} = a(\mathbf{W}\mathbf{h}_i, \mathbf{W}\mathbf{h}_j)$ は，学習可能な重み行列 $\mathbf{W} \in \mathbb{R}^{F'\times F}$ を用いて線形変換を行い，Attention mechanism ($a : \mathbb{R}^{F'} \times \mathbb{R}^{F'} \to \mathbb{R}$) によって算出される．$e_{ij}$ は，ノード v_i におけるノード v_j の重要度を表す．隣接ノードの重要度は，以下の式で定義できる．

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k\in\mathcal{N}_i}\exp(e_{ik})} \tag{7}$$

重要度を表す α_{ij} は，見出しセルの識別タスクにおいて，ラベルを予測するうえで重要なセル間の関係性を評価したもので，予測精度の向上に寄与するような関係性に対しては大きい値，ほとんど無関係な関係性には小さい値がかかる．見出し階層の認識タスクにおいても階層関係の予測精度が向上する関係性に対して重要度が高くなる．

上記の式を用いて，アテンションによる特徴行列の更新は，以下の式で定義できる．

$$\mathbf{h}'_i = \sigma\left(\sum_{j\in\mathcal{N}_i}\alpha_{ij}\mathbf{W}\mathbf{h}_j\right) \tag{8}$$

さらに，学習過程におけるアテンション係数の安定化のために，マルチヘッドなアテンション機構を取り入れる．

$$\mathbf{h}'_i = \|_{k=1}^{K}\sigma\Big(\sum_{j\in N_i}\alpha_{ij}^k\mathbf{W}^k\mathbf{h}_i\Big) \tag{9}$$

ただし，K 個の独立したアテンション機構の導入を意味する $\|_{k=1}^{K}$ は，それぞれで更新された特徴表現を特徴ベクトルの方向へ結合する演算を表す．この場合，特徴ベクトルが F' 次元で K 個結合すると次元数は $F'K$ になる．最後に，拡大したノード特徴ベクトル（$F'K$ 次元）について，出力がラベルの次元数と等しくなるような適切な重みを持つアテンション機構を加えることで最終的な出力を得る．ここでは，特徴ベクトルの結合ではなく，平均を取る．

3.4 事前学習によるノード埋め込みによる手法

グラフニューラルネットワークによる教師なし表現学習手法の Deep Graph Infomax (DGI) [3] について述べる．

DGI は，Deep Infomax (DIM) [13] と呼ばれる教師なし表現学習手法をグラフデータに適用することで,「グラフ上の局所領域の特徴（パッチ表現）と大域的な特徴の間における相互情報量の最大化」が可能となり，局所的に見れば意味のないノード特徴表現よりも統計表全体に関連のある特徴表現を学習過程で得られるようにすることができる．DIM は，エンコーダの中間の畳み込み層から得られた特徴マップの各ピクセル毎と出力層から得られる特徴表現を用いて対照推定を行い，入力と特徴表現の相互情報量の最大化を行う．

DGI の目標は，$\mathcal{E}(\mathbf{X}, \mathbf{A}) = \mathbf{H}$ が各ノード v_i の大域的な特徴表現 $\mathbf{h}_i \in \mathbb{R}^{F'}$ を表すようなエンコーダ $\mathcal{E} : \mathbb{R}^{N\times F} \times \mathbb{R}^{N\times N} \to \mathbb{R}^{N\times F'}$ を学習することである．入力には，ノードの特徴行列 $\mathbf{X} \in \mathbb{R}^{N\times F}$ と隣接行列 $\mathbf{A} \in \mathbb{R}^{N\times N}$ が与えられ，学習後にノードの埋め込み $\{\mathbf{h}_n \in \mathbb{R}^{F'}\}_{n=1}^{N}$ ($F' \ll N$) が得られる．これらのノード埋め込みは，ノード分類などの下流タスクで利用できる．学習過程のおけるノードの潜在特徴ベクトルは，GCN などをエンコーダとして採用することで，局所的なノード近傍 $\mathcal{N}(v)$ に対して繰り返し情報を集約し，自身の特徴ベクトルと足し合わせ更新される．そのため，ノード埋め込み $\mathbf{h}_i$ は自身を中心としたグラフのパッチ表現を要約している．

エンコーダの学習では，局所的な相互情報量を最大化する，つまり，大域的な特徴ベクトルで表されるグラフ全体のグローバルな情報内容を捉えられるノード（すなわち，局所的な）特徴表現を求めている．また，グラフ全体の情報内容を表す，グラフレベルの要約ベクトル $\mathbf{s}$ に関しては，読み出し関数 $\mathcal{R} : \mathbb{R}^{N\times F} \to \mathbb{R}^F$ で得られるパッチ表現 $\mathbf{s} = \mathcal{R}(\mathcal{E}(\mathbf{X}, \mathbf{A}))$ を使用する．ここでの読み出し関数 $\mathcal{R}$ は，エンコーダから出力される局所的な特徴ベクトルをグラフレベルの特徴ベクトルとして生成する関数である．

局所的な相互情報量を最大化するには，正例のペアと負例のペアを区別するように識別器 $\mathcal{D} : \mathbb{R}^F \times \mathbb{R}^F \to \mathbb{R}$ を学習させる．正例は，エンコードされたノード潜在特徴ベクトル $\{\mathbf{h}_n\}_{n=1}^{N}$ と読み出し関数 $\mathcal{R}$ から得られるパッチ表現 $\mathbf{s}$ をペアとするデータ $\{(\mathbf{h}_1, \mathbf{s}), (\mathbf{h}_2, \mathbf{s}), \cdots, (\mathbf{h}_N, \mathbf{s})\}$ で，負例は異なるデータ $(\tilde{\mathbf{X}}, \tilde{\mathbf{A}})$ からエンコードされたノード潜在特徴ベクトル $\{\tilde{\mathbf{h}}_m\}_{m=1}^{M}$ とパッチ表現 $\mathbf{s}$ をペアとするデータ $\{(\tilde{\mathbf{h}_1}, \mathbf{s}), (\tilde{\mathbf{h}_2}, \mathbf{s}), \cdots, (\tilde{\mathbf{h}_M}, \mathbf{s})\}$ とする．M の値は N と同じとする．負例のためのデータ $(\tilde{\mathbf{X}}, \tilde{\mathbf{A}})$

の選択としては，元と同じ隣接行列 $\tilde{\mathbf{A}} = \mathbf{A}$ のデータから $\tilde{\mathbf{X}}$ をランダムにシャッフルする方法などがある．各ノード表現とグローバル表現の間の相互情報量を最大化するために目的関数は，同時確率分布と周辺分布の積が区別できるように，周辺分布からの正例と負例の間の Binary Cross Entropy (BCE) となる．

$$\mathcal{L} = \frac{1}{N+M}\left(\sum_{i=1}^{N}\mathbb{E}_{(\mathbf{X},\mathbf{A})}\left[\log \mathcal{D}\left(\mathbf{h}_i, \mathbf{s}\right)\right] + \sum_{j=1}^{M}\mathbb{E}_{(\tilde{\mathbf{X}},\mathbf{A})}\left[\log\left(1-\mathcal{D}\left(\mathbf{h}_j, \mathbf{s}\right)\right)\right]\right) \tag{10}$$

得られるパッチ表現は，グラフレベルの要約ベクトルとの相互情報量を保持するように学習されるため，パッチレベルでの類似性の発見が可能となり，見出し階層の識別タスクに有効な特徴表現であると考えられる．以上より，各ノードの表現とグローバル表現の間の相互情報量を最大化するという問題は，正例と負例の間の識別器を学習する問題として帰着される．

4 実　　験

我々のタスクに適した公開済みのデータセットが存在しないため，まずデータセットの作成の概略と統計情報について説明する．さらに，ベースライン手法を含む実験設定について紹介し最後に実験結果を示す．

4.1 データセット

本研究では，見出しとその階層関係の認識タスクのために Excel 形式の統計表データを「政府が実施する統計調査の結果をオープンデータとして公開している政府統計の総合窓口 (e-Stat)[3] 」から収集されたデータセット [14] を使用した．

収集した Excel データは，前節で述べた通り，グラフデータに変換して使用するため，各ノードの特徴ベクトルを作成する必要があった．特徴ベクトルを構成する複数の要素は,「セル値」と「空白数」,「文字サイズ」,「文字フォントの様式情報」,「セル種別の有無」,「罫線情報」,「セルの高さと幅」とした．特徴ベクトルに変換するために，文字フォントの様式情報は，太字と斜字の有無 $\{0,1\}$ を，罫線情報は，セルの上下左右に位置する罫線の 4 パターンとセル内の右下斜めに引かれた罫線の有無 $\{0,1\}$ を用いた．また，セル種別の有無 $\{0,1\}$ はセル値に基づく種別判定を行い，未記入なら空白ラベルを 1，値が数値なら，数値ラベルを 1 に，セル同士が結合なら結合ラベルを 1 として，それ以外なら 0 とした．これらの属性は，通常の One Hot Encoding ではなく Binary Encoding を行い，量的変数に変換した．数値属性としては,「文字サイズ」と「セル幅と高さ」が該当し，抽出後に各属性ごとに $[0,1]$ に収まるよう各属性の最大値によって正規化を行った．セル値の分散表現は，事前学習済みの単語分散表現モデル fastText[4] [15] より獲得した．

我々は，本タスクのためにスプレッドシートを対象としたアノテーションツールの開発を行い，見出しセルと見出し階層に

3：https://www.e-stat.go.jp/

4：https://fasttext.cc/docs/en/crawl-vectors.html

	総数	2人以上の世帯			
		総数	世帯主	世帯主の配偶者	その他の家族
正規の職員・従業員	3534	2922	1522	511	888
非正規の職員・従業員	2045	1741	428	820	492
パート・アルバイト	1407	1230	204	675	351
パート	986	887	136	615	137
アルバイト	421	343	69	60	214

見出しセル　数値セル　見出しの階層関係

図 1　スプレッドシート内の統計表の例

表 1　データセットの統計情報と正解ラベルの内訳

	事前学習データ	教師データ
# Graphs	100,000	44
# Nodes	77,231,159	53,929
# Edges	495,615,486	347,068
# Features	314	314
# Nested Headers (Postive)		1,304 (6.407 %)
# Nested Headers (Negative)		19,049 (93.593 %)
# Headers (Postive)		3,059 (5.672 %)
# Headers (Negative)		50,870 (94.328 %)
# Bold Fonts (Postive)	799,027 (1.035 %)	129 (0.239 %)
# Bold Fonts (Negative)	76,432,132 (98.965 %)	53,800 (99.761 %)
# Italic Fonts (Postive)	12,803 (0.017 %)	0 (0.000 %)
# Italic Fonts (Negative)	77,218,356 (99.983 %)	53,929 (100.000 %)
# Empty Cells (Postive)	33,126,246 (42.892 %)	30,199 (55.998 %)
# Empty Cells (Negative)	44,104,913 (57.108 %)	23,730 (44.002 %)
# Merged Cells (Postive)	1,206,714 (1.562 %)	781 (1.448 %)
# Merged Cells (Negative)	76,024,445 (98.438 %)	53,148 (98.552 %)
# Numerical Cell (Postive)	28,795,295 (37.285 %)	14,882 (27.596 %)
# Numerical Cell (Negative)	48,435,864 (62.715 %)	39,047 (72.404 %)

あるセルペアの正解ラベルを 44 ファイルに付与した．シート内の統計表における見出しセルとその階層関係のラベルを含んだ例を図 1 に示す．この統計表には左側と上側の両方に見出し階層関係が存在しており，左側の見出し階層では空白セルを文章でのインデント（字下げ）として利用することで，その階層を表現している．具体的には，空白セルをシート内に挿入することで開始位置を右に移動させ，その上に位置する見出しに対して階層をもつ見出しセルであることを示している．また，上側の見出し階層では，複数のセルで結合された見出しセルに対してその下に位置する見出しが階層関係をもっている．

表 1 に統計情報と正解ラベルの内訳を示す．見出し階層のセルペア (# Nested Headers) と見出しセル (# Headers) の正解ラベルは，どちらも不均衡なデータとなっており，全体のセルの中で見出しである（正例の）割合は低く，同様にセル同士の関係で見出し階層関係にある割合も低いことがわかった．また，シート内の空白セル (# Empty Cells) である割合は数値セル (# Numerical Cells) である割合よりも高くなっており，空白セルである割合は高いことが分かる．結合セル (# Merged Cells) である割合は全体の 1% である．文字フォントの様式情報に関しては，太字 (# Bold Fonts) である割合は全体の 1% を占めるが，斜字 (# Italic Fonts) である割合はそれにも満たない．

表 2　見出し**階層**の認識タスクの分類結果.

	GCN [11]	GAT [12]	Log-Reg	Li-GBM [16]	Log-Reg (DGI)	Li-GBM [16] (DGI)
F1	**0.646**	0.625	0.565	0.502	0.630	0.568
Accuracy	**0.950**	**0.950**	0.928	0.946	0.944	0.948
Precision	0.588	0.600	0.460	**0.619**	0.543	0.607
Recall	0.717	0.653	0.733	0.423	**0.750**	0.534
True Positive	935	851	956	551	**978**	696
True Negative	18,395	18,481	17,927	**18,710**	18,226	18,598
False Positive	654	568	1,122	**339**	823	451
False Negative	369	453	348	753	**326**	608

表 3　見出し**セル**の識別タスクの分類結果.

	GCN [11]	GAT [12]	Log-Reg	Li-GBM [16]	Log-Reg (DGI)	Li-GBM [16] (DGI)
F1	0.732	0.844	0.862	**0.872**	0.657	0.720
Accuracy	0.967	0.982	0.984	**0.985**	0.960	0.968
Precision	0.671	0.823	0.859	**0.872**	0.635	0.714
Recall	0.805	0.866	0.865	**0.871**	0.681	0.726
True Positive	2,461	2,648	2,647	**2,665**	2,084	2,221
True Negative	49,666	50,300	50,434	**50,479**	49,671	49,982
False Positive	1,204	570	436	**391**	1,199	888
False Negative	598	411	412	**394**	975	838

4.2 実験設定

我々は実験における比較手法として，GNN による手法と DGI の下流タスクで使用する手法（初期特徴ベクトルを使用）を用いた：(1) **GCN** [11]: 隣接ノードに対して畳み込み演算を行いクロスエントロピー損失を最適化する手法，(2) **GAT** [12]: 隣接ノードに対して重要度の計算を行いクロスエントロピー損失を最適化する手法，(3) **Logistic Regression(Log-Reg)**: 線型回帰の従属変数の値をロジット変換で $[0, 1]$ の値に変換し二値化する手法，(4) **Light GBM(Li-GBM)** [16]: 決定木アルゴリズムに基づいた勾配ブースティング（Gradient Boosting）の機械学習手法.

予測モデルの汎化性能を正確に評価するために 5 分割交差法を採用し，構築したデータセットを用いた本タスクの手順を以下で述べる．(1) 全データを 5 個に分割し，(2) 分割したデータの 1 つを評価用とし，残りのデータのうち 1 つを検証用として，それ以外で学習を行い，(3) 評価用データで予測値 $\{0, 1\}$ を出力し，(4) 全データの予測値が出力できるまで，手順 2，3 を繰り返す.

ハイパーパラメータチューニングでは，すべてのパラメータの組み合わせの中で，検証用データの分類精度が最も高くなる組み合わせを探索し，各手法の最適なパラメータを決定する．ただし，二値分類問題の評価指標には F1 値 (F1-score) を使用する．教師なし表現学習 (**DGI**) のパラメータ設定では，隠れ層の潜在特徴ベクトルの次元数を 512，エポック数を 10 として，事前学習データに適用した．その後，学習済みのモデルから得られるノード埋め込みを入力とする分類器を教師データに適用し，5 分割交差法で分類精度を算出する．分類器の手法としては，Log-Reg と Li-GBM を採用する.

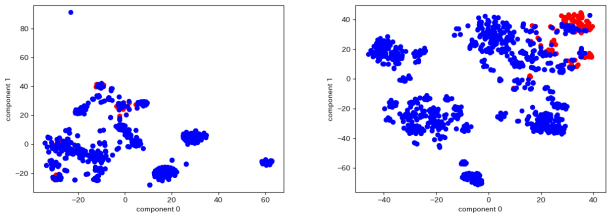
4.3 実験結果

表 2 に見出し階層の認識タスクの分類結果を示す．教師なし表現学習（DGI）で得られたノード埋め込みを用いた分類器の F1 値が，初期特徴ベクトルを用いた分類器よりも高くなっており，このタスクに有効な特徴表現を事前学習で獲得していると考えることができる．しかし，GNN の教師あり学習手法 (GCN) と比較して F1 値が低くなっていることから，教師なし表現学習手法における事前学習のさらなる調整が必要であると言える．Li-GBM の F1 値は他の手法と比べて低くなっており，この手法は目的変数の分布に大きな偏りがある不均衡データに対して，分類精度が向上できないと判明した．反対に，Log-Reg は不均衡データに対しても分類性能を発揮できている．また，GNN の教師あり学習手法が見出し階層の認識タスクにおいて効果的であることが明らかになり，GNN の特徴である「グラフにおけるノード情報を伝搬することでノード自身の潜在特徴ベクトルを更新する」という学習方法が見出し階層の認識に有効であると考えられる.

シート内の見出し階層があるセルペアを適切に識別するためには，適合率（precision）の向上が課題である．F1 値が最も高かった GCN でも階層関係が存在すると判定した件数のうち約 4 割が不正解となっており，タスクの難易度が高いことが分かる．人為的な正解データを含めた教師データの増強が，表構造の意味解釈を適切に行うために重要になると考えられる．人為的な正解データとは，実在する統計表データの中で複雑な見出し階層が存在するデータを複製する，もしくは人手で類似のデータを作成することを指し，正例の割合を増やすために有力な解決策であると思われる.

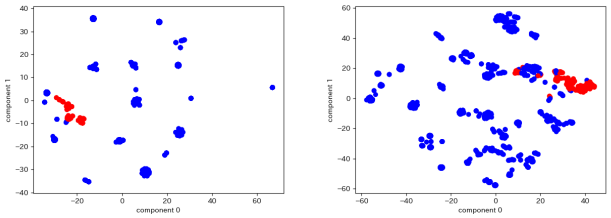
表 3 に見出しセルの識別タスクの分類結果を示す．教師なし表現学習（DGI）で得られたノード埋め込みを用いた分類器の F1 値は見出し階層の認識タスクとは異なり，初期特徴ベクトルを用いた分類器よりも低くなっていることがわかった．このタスクに有効な特徴表現を獲得できていないと考えられる．また，GNN の教師あり学習手法は分類器で用いた手法と比較して F1 値が低くなっており，グラフ構造データの特性を活かした情報伝搬が悪影響を及ぼしている可能性があると思われる．初期特徴ベクトル，言い換えれば，シート内から取得できるセル情報を用いた分類器の F1 値が 0.862，0.872 という評価であることから，ニューラルネットワークを用いなくても優れた分類精度を得られることが明らかになった．ただし，グラフ内の隣接ノードの重要度を加味するアテンション機構を用いた GAT の F1 値は最も高い値と比べて，-0.028 の差異であるため畳み込み演算を変更すれば，GNN の精度が改善される可能性もあると思われる.

DGI で得られる特徴表現（埋め込み）を入力とした分類器の精度だけでなく，高次元の特徴表現が分類問題を解く上で有



(a) raw features　　(b) DGI features

図 2　t-SNE を用いた次元圧縮による可視化（見出し**階層**の認識タスク）．データ点は赤色を正例，青色を負例として示している．



(a) raw features　　(b) DGI features

図 3　t-SNE を用いた次元圧縮による可視化（見出し**セル**の識別タスク）．データ点は赤色を正例，青色を負例として示している．

用かどうかを調べる．t-SNE [17] を利用して，高次元のノード埋め込みを次元圧縮することで 2 次元空間上に可視化する．具体的には，高次元空間上の x_i, x_j の類似度と低次元空間上の y_i, y_j の類似度をそれぞれ確率分布 p_{ij} と q_{ij} で表現し，2 つの確率分布の差が最小となるように低次元への変換が行われる．教師データを次元圧縮することで高次元空間におけるデータ同士の「近さ（類似度）」が低次元空間に反映され，ノード埋め込みの有効さを調べることができる．

図 2，3 に t-SNE を用いた次元圧縮による見出し階層のセルペアと見出しセルの埋め込みを可視化した結果をそれぞれ示す．見出し階層のセルペアの埋め込みは，DGI の教師なし学習（事前学習）によって分離された埋め込みが生成されているが，多数の負例も混在しているため分類性能が向上しづらいと言える．見出しセルの埋め込みについては，初期特徴ベクトルの方が十分に分類されている．そのため，事前学習の効果が見られないと判断できる．この可視化で埋め込みの有効性を見極めることが可能であると分かり，見出しセルの識別タスクにおいて初期特徴ベクトルを入力とした分類器の精度が高くなることも容易に調べられることも判明した．

最後に，少量の教師データで学習する条件下で，初期特徴ベクトルと DGI の埋め込みの異なる入力ベクトルで分類精度に違いが生まれるかを検討する．この実験で学習データ内に多く存在しない傾向のデータ（見出しとその階層の正例）に対して適切に予測を行うために DGI の埋め込みが有効であるかどうかを明らかにしたいと考えている．また，このような多様なデータがある中で適切に予測を行うことができる能力のことを機械学習における頑健性（ロバストネス）という．本データセットも正例の正解データが限られた状況下で適当な予測を行える「頑健」なモデルを構築することが求められる．検討方法としては，本実験と同じ不均衡な正解ラベルが付与されたデータセットの中で学習データを減らしていき，実験と同様の 5 分割交差法で分類結果を出力する．期待される結果として，1 つ目は，「(A) DGI の埋め込みであれば，少量の教師データでも分類器の精度が他の手法と比べて安定し易い」とし，2 つ目は，「(B) 学習データが減少し全体で約 6% の正例事例がさらに少なくなる状況で事前学習が役立つのではないか」とする．

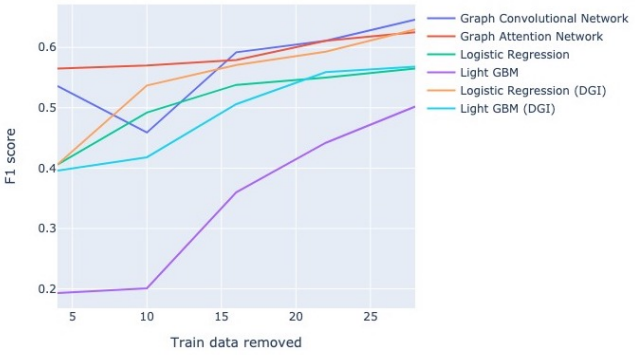


図 4　見出し**階層**の認識タスクにおける手法の頑健性の評価結果．実験と同一の条件下で学習データ数を減少させて，テストデータの予測結果の F1 値を示している．

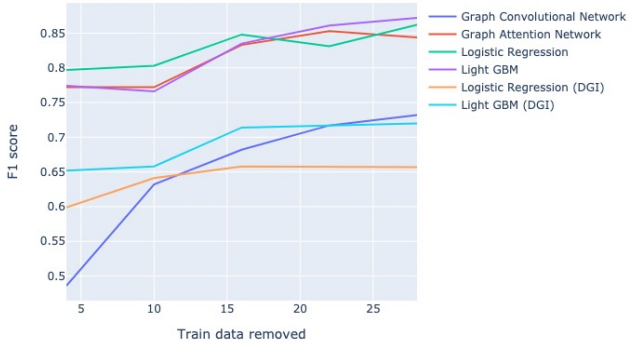


図 5　見出し**セル**の識別タスクにおける手法の頑健性の評価結果．実験と同一の条件下で学習データ数を減少させて，テストデータの予測結果の F1 値を示している．

教師データを段階的に減少させながら学習させた手法の分類結果を図 4，5 に示す．見出し階層の認識タスクにおける手法の頑健性の評価では，GAT が最も安定していることがわかった．グラフ内に空白セルが多数存在する中でノードに重要度を付ける方法が良いと考えられる．GCN も同じ GNN の手法であるが，10 個の教師データの時には F1 値が急激に減少していた．DGI の埋め込みを入力とする分類器の中で，Log-Reg は 15 個の教師データになるまで F1 値が GCN と GAT と競合していた．また，5 個の学習データになると急激に F1 値が下がってしまう事実もあることは注意すべき点であると思われる．見出しセルの識別タスクにおける頑健性の評価では，Log-Reg が最も安定していることがわかった．Li-GBM と GAT の F1 値は同じような推移をしており，Log-Reg は少量の教師データでも最も安定した性能を発揮できるとわかった．GCN の F1 値は 20 ～ 10 個に学習データを減少させる過程で最も急激に下がっており，安定した分類性能に向けた学習が困難な手法であ

るとわかった．そのことから，仮説 (A) のみ正しく，仮説 (B) は正しくない．また，正例の正解データが限られた状況下で適当な予測を行える「頑健」なモデルとして見出し階層の認識タスクは GAT，見出しセルの識別タスクでは Log-Reg であるとわかった．GAT は見出しセルの識別タスクにおいても一定の精度を保っており，見出しセルかどうかを区別するために重要度を付ける方法が有効であると考えられる．

5 まとめ

本論文では，スプレッドシート内の統計表を構成するセルの視覚的な情報（文字列や値，罫線など）をノード特徴とするグラフデータに変換し，グラフニューラルネットワーク (GNN) を適用することによって，スプレッドシートの大域的な特徴を考慮して統計表における見出しとその階層関係の認識（見出し認識）を行う方法を提案した．また，見出し認識の教師データを用意するためには大きな労力が必要となるため，教師なし表現学習を行うことで認識タスクに有効なセル表現を学習し，少数の教師データであっても効果的な学習が可能となる方法を提案した．実験では，見出し階層のセルペアの認識と見出しセルの識別を対象とし，比較手法を従来の機械学習による手法として，事前学習済みの DGI から得られる埋め込みを用いて分類器の評価を行った．データセットには，政府統計ポータルサイト e-Stat から収集されたスプレットシートを使用し，認識タスクの正解ラベルを用意するためにアノテーションを行った．

実験の結果，GNN の教師あり学習手法が従来の機械学習による手法よりも見出し階層の認識で高い精度を達成できることを示した．今後の課題としては，正例ラベルの割合を増やすことを目的とした教師データの作成，教師なし表現学習における事前学習の改善などが挙げられる．

謝辞 本研究は JSPS 科研費 18H03244，18H03243，および，JST さきがけ JPMJPR1853 の助成を受けたものです．ここに記して謝意を表します．

文　献

[1] Majid Ghasemi Gol, Jay Pujara, and Pedro Szekely. Tabular cell classification using pre-trained cell embeddings. In *2019 IEEE International Conference on Data Mining*, pages 230–239, Beijing, China, 2019-11-8/11, 2019. IEEE.

[2] 曽和修平, 宮森恒. 複雑な構造をもつ統計表における見出し階層の認識と意味解釈手法. 2017, (B4-4).

[3] Petar Veličković, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. Deep Graph Infomax. In *2019 7th International Conference on Learning Representations*, New Orleans, LA, USA, 2019-05-6/9, 2019. OpenReview.net.

[4] 松井侑祐, 宮森恒. 統計表データを用いた動向情報の根拠探索システムの検討. 2014.

[5] 松井侑祐, 宮森恒. 動向情報の根拠探索を目的とした統計表データの自動認識. 2015, (B4-5).

[6] Elvis Koci, Maik Thiele, Wolfgang Lehner, and Oscar Romero. Table Recognition in Spreadsheets via a Graph Representation. In *2018 13th IAPR International Workshop on Document Analysis Systems*, pages 139–144, Vienna, Austria, 2018-04-24/27, 2018. IEEE Computer Society.

[7] Elvis Koci, Maik Thiele, Óscar Romero Moral, and Wolfgang Lehner. A Machine Learning Approach for Layout Inference in Spreadsheets. In *Proceedings of the International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*, pages 77–88, Porto, Portugal, 2016-11-9/11, 2016. SciTePress.

[8] Felienne Hermans and Emerson Murphy-Hill. Enron's Spreadsheets and Related Emails: A Dataset and Analysis. In *2015 37th IEEE/ACM IEEE International Conference on Software Engineering*, pages 7–16, Florence, Italy, 2015-05-16/24, 2015. IEEE Computer Society.

[9] Titus Barik, Kevin Lubick, Justin Smith, John Slankas, and Emerson Murphy-Hill. Fuse: a reproducible, extendable, internet-scale corpus of spreadsheets. In *2015 12th IEEE/ACM Working Conference on Mining Software Repositories*, pages 486–489, Florence, Italy, 2015-05-16/17, 2015. IEEE Computer Society.

[10] Marc Fisher and Gregg Rothermel. The EUSES spreadsheet corpus: a shared resource for supporting experimentation with spreadsheet dependability mechanisms. In *Proceedings of the First Workshop on End-User Software Engineering*, pages 1–5, Saint Louis, Missouri, USA, 2005-05-21, 2005. ACM.

[11] Thomas N. Kipf and Max Welling. Semi-Supervised Classification with Graph Convolutional Networks. In *2017 5th International Conference on Learning Representations*, Toulon, France, 2017-04-24/26, 2017. OpenReview.net.

[12] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph Attention Networks. In *2018 6th International Conference on Learning Representations*, Vancouver, BC, Canada, 2018-04-30/05-03, 2018. OpenReview.net.

[13] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. In *2019 7th International Conference on Learning Representations*, New Orleans, LA, USA, 2019-05-6/9, 2019. OpenReview.net.

[14] Makoto P Kato, Hiroaki Ohshima, Ying-Hsang Liu, and Hsin-Liang Chen. Overview of the ntcir-15 data search task. *Proceedings of the NTCIR-15 Conference*, pages 267–273, 2020.

[15] Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. Learning word vectors for 157 languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation*, Miyazaki, Japan, 2018-05-7/12, 2018. ELRA.

[16] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A Highly Efficient Gradient Boosting Decision Tree. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, pages 3146–3154, Long Beach, CA, USA, 2017-12-4/9, 2017.

[17] Laurens van der Maaten and Geoffrey E. Hinton. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008.