DEIM2021 E25-3

ニューラルネットワークを用いた表構造解析の一手法

青柳　拓志† 　金澤　輝一†† 　高須　淳宏†† 　上野　史††† 　太田　学†††

† 岡山大学工学部情報系学科　〒 700–8530 岡山県岡山市北区津島中 3-1-1
†† 国立情報学研究所　〒 101–8430 東京都千代田区一ツ橋 2-1-2
††† 岡山大学大学院自然科学研究科　〒 700–8530 岡山県岡山市北区津島中 3-1-1
E-mail: †ao2516@s.okayama-u.ac.jp, ††{tkana，takasu}@nii.ac.jp,
†††uwano@okayama-u.ac.jp, ohta@cs.okayama-u.ac.jp

あらまし　学術論文では，実験結果をまとめるのに表がしばしば用いられるが，多くの実験結果を一目で把握するには視覚的に優れたグラフが適している．そのため，表からグラフを自動生成する研究が行われているが，表の書き方は著者によって異なるためまず表の構造を解析する必要がある．そのため本稿では，ニューラルネットワークを用いた表構造解析手法を提案する．提案手法は，表中のトークンをマージするニューラルネットワーク（NN），実際には引かれていないがセルを分割するのに必要な罫線（補助罫線）を推定する NN，表中のトークンをマージすることによりセルを生成する NN，の三つを用いて表構造を解析する．また提案手法は，入力として表の PDF 文書だけでなく表画像も構造解析する．実験では，International Conference on Document Analysis and Recognition 2013(ICDAR 2013) の Table competition で提供されたテスト用文書 PDF を用いて表構造の解析精度を評価した．実験の結果，セルの隣接関係に基づく解析精度を示す評価指標で，再現率が 0.967，適合率が 0.977，F 値が 0.972 となり，2020 年に発表された山田らの手法をそれぞれ 1.6 ポイント，1.7 ポイント，1.7 ポイント上回った．また同じデータの表画像を入力として構造解析すると，表を HTML の木構造で表した類似度で 0.831 となった．このとき同じ表の元の PDF を入した場合，その類似度は 0.956 だった．

キーワード　表構造解析，グラフ自動生成

1 はじめに

近年，CiNii[1]や Google Scholar[2] といった学術論文データベースの発展により膨大な学術論文が容易に入手できるようになった．学術論文では，実験結果を示すのに表が頻繁に用いられる．しかしながら，多くの実験結果を効率良く把握するには視覚的に優れたグラフのほうが適している．そこで表のグラフへの自動変換のため，山田ら [1] は実際には引かれていないがセルを分割するのに必要な線分である補助罫線に着目した表構造解析手法を提案した．

山田らの手法は，表中のトークンの位置関係に基づいて補助罫線を推定するニューラルネットワーク（NN）と，推定された補助罫線やトークンの特徴等を用いてセルを生成する NN を用いて表構造を解析する．彼らは，ICDAR 2013 の Table competiton [2] で提供されたテスト用データセットの表を構造解析し，表中のセルの隣接関係の再現率が 0.951，適合率が 0.960，F 値が 0.955 となり，ICDAR 2013 の Table competiton の参加者の最高結果を上回った．

本稿では，山田らの手法を改良したニューラルネットワークを用いた表構造解析手法を提案する．とりわけ，英語で書かれた表は多くのトークンで構成されるセルが多いことに着目する．提案手法では，補助罫線推定前に表中のトークンをマージする NN，補助罫線を推定する NN，表中のトークンをマージすることによりセルを生成する NN，の三つの NN を用いて表構造を解析する．さらに，山田らは扱わなかった，表画像を入力とする表構造解析も行う．実験では，PDF 中の表の表構造解析には ICDAR 2013 の Table competiton で提供されたテスト用データセット，表画像の表構造解析にはそのテスト用データセットの PDF を画像に変換して抽出した表を使って評価する．評価指標にはセルの隣接関係に基づく指標等を用いる．

2 関連研究

山田らは，機械学習を用いた表構造解析手法を提案した [1]．彼らの手法の概要を図 1 に示す．彼らの手法では，まず pdfalto[3]を用いて文書 PDF を XML ファイルへ変換する．また，PDFMiner[4] と OpenCV[5]を用いて表中の罫線を検出する．次に，XML ファイルから得られる表中のトークンの位置関係に基づいて実際には引かれていない罫線である補助罫線を推定する．つづいて，罫線，推定した補助罫線，隣接 2 トークンの特徴，周辺のトークンの特徴，トークンの分散表現を用いてセルを生成する．最後に，行や列の結合とセルの拡張を行い，

1 : https://ci.nii.ac.jp
2 : https://scholar.google.co.jp
3 : https://github.com/kermitt2/pdfalto
4 : https://github.com/pdfminer/pdfminer.six
5 : https://opencv.org

図 1　山田らの表構造解析手法

最終的な表構造を確定する．ICDAR2013 で開催された Table competition にて提供されたテスト用データセットを評価実験に用いた結果，セル間の隣接関係の再現率が 0.951，適合率が 0.960，F 値が 0.955 となり，ICDAR 2013 の表構造解析コンペティションのすべての参加者の結果を上回った．本稿は，この手法を改良する．

Qasim らは，グラフニューラルネットワークを用いた表構造解析手法を提案した [3]．この手法では，まず畳み込みニューラルネットワークの入力として表画像を与え，畳み込み特徴を得て，畳み込み特徴量を頂点特徴へ拡張する．次に，interaction model に頂点特徴を入力として与え，分類に使用する代表特徴を出力として得る．なお，interaction model にはグラフニューラルネットワークを用いる．最後に，代表特徴をニューラルネットワークに与え，予測したクラスを出力として得る．Qasim らは，実験の結果からグラフニューラルネットワークは，複数行や複数列にまたがるセルを含む表には頑健でないが，矩形でないセルを含む表はうまく解析できると結論づけた．

Paliwal らは，表検出と表構造認識の間の固有の相互依存性に着目したエンドツーエンドのディープニューラルネットワークである TableNet を提案した [4]．このモデルは，エンコーダ，表を検出するデコーダと表の列を検出するデコーダを持つ．まず，画像を入力として与え，表とその列を検出する．次に，ルールベースで行を抽出する．最後に，それらの結果を利用してセルの内容を抽出する．実験では，ICDAR 2013 データセットを用いて評価しその結果，ディープラーニングに基づく表構造解析手法である DeepDSert [5] の結果をわずかに上回った．

Zhong らは，encoder-dual-decoder(EDD) を用いた表構造解析手法を提案した [6]．EDD は，エンコーダ，構造デコーダとセルデコーダから構成されている．エンコーダには畳み込みニューラルネットワークを用い，構造デコーダとセルデコーダには attention 付きのリカレントニューラルネットワークを用いた．Zhong らは，表構造の認識とセルの内容の認識は独立に区別できるタスクで，表構造の情報はセルの内容の認識に有用であると考えた．Zhong らの手法では，まずエンコーダで入力表画像の視覚的特徴をとらえる．次に，構造デコーダで表構造を定義する HTML タグを生成する．また，構造デコーダが新たにセルを認識した場合に，セルデコーダは構造デコーダの隠れ状態を用いて attention を計算しそのセルの内容を認識する．最後に，構造デコーダの結果とセルデコーダの結果をマージして最終的な表の HTML コードを生成する．モデルの学習には，PubTabNet[6] を用いた．実験では，複数の行や列にまたがるセルが存在しない単純な表のデータセットと複数の行や列にまたがるセルが存在する複雑な表のデータセットを用いて表構造解析精度を評価した．評価指標には，HTML 形式で表した表の木構造としての類似度を定義した tree-edit-distance-based similarity(TEDS) を用いた．その結果，TEDS で単純な表のデータセットで 0.912，複雑な表のデータセットで 0.854 となり，WYGIWYS [7] の結果をそれぞれ 9.5 ポイント，9.9 ポイント上回った．

3　提 案 手 法

3.1　概　　要

図 2 に本稿の提案手法の概要を示す．まず，文書 PDF または表画像を入力として与える．次に，前処理で文書 PDF または表画像の表中のトークンと罫線を取得する．PDF 入力の場合は，山田らの手法と同様に pdfalto を用いて文書 PDF を XML 化してトークンを取得し，PDFMiner と OpenCV を用いて罫線を取得する．表画像の場合は，tesseract[7] を用いて表中のトークンを取得し，OpenCV で表中の線分を検出する．具体的には，まず表の端から端までまたがるような長い線分を検出しその線分を削除する．これは，罫線が OCR の認識精度に悪影響を与えるためである．そして，その線分が削除された表画像からトークンを取得する．最後に，再び線分を検出する．これは，最初の線分の検出で検出されなかった短い罫線を検出するためである．なお，取得した線分の内，トークンと重なっているものは罫線ではなく文字や単語の一部分であるため削除し，残ったものを罫線とする．つづいて，トークンの特徴を用いて，水平方向に隣接する 2 トークンを再帰的にマージする．マージし終えたら，実際には引かれていないがセルを分割するのに必要な線分である補助罫線をトークンの位置関係に基づいて推定する．そして，トークンの特徴と，罫線・補助罫線の情報を用いて，垂直方向に隣接する 2 トークンを再帰的にマージする．マージし終えたら隣接 2 トークンの水平マージと垂直マージを交互に再帰的に行ってセルを生成する．最後に，山田らの手法と同様に後処理で行や列の結合とセルの拡張を行い，最終的な表構造を確定する．

3.2　本稿が構造解析の対象とする表の構成要素

図 3 に本稿で扱う表の例を示す．なお，赤い矩形，点線は本来の表にはなく，説明のため便宜的に加えたものである．図 3 中の実線が罫線，点線が補助罫線である．また，赤い矩形で囲まれたものをトークンと呼び，罫線または補助罫線で囲ま

6 : https://github.com/ibm-aur-nlp/PubTabNet
7 : https://github.com/tesseract-ocr/tesseract

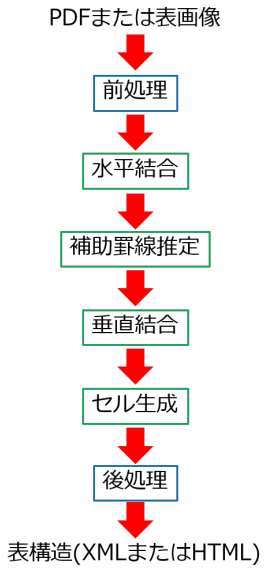

図 2　提案する表構造解析手法の概要

	Recall	Precision
Method A	0.92	0.83
Method B	0.90	0.87
Method C	0.96	0.92

図 3　表と表の構成要素

れているものをセルと呼ぶ．なお，トークンは pdfalto または tesseract で取得し，表中の単語であることが多い．

3.3　水 平 結 合

3.3.1　概　　要

水平結合は，表中の水平方向に隣接する 2 トークンの特徴と周辺のトークンの特徴を用いて，その 2 トークンをマージするものである．なお，マージする水平方向に隣接するトークンがなくなるまでマージをつづける．水平結合の NN のモデル図を図 4 に示す．図 4 のモデルへの入力は，107 次元の水平方向に隣接する 2 トークンの特徴ベクトル (Two adjacent tokens' features) と 144 次元の周辺のトークンの特徴ベクトル (Surrounding tokens' features) である．また，出力はマージする確率としない確率の 2 次元である．中間層の出力次元数は，連結層より前で周辺のトークンの特徴ベクトルを入力とする層は 20，結合層より後では 300 とする．出力層の活性化関数は Sigmoid 関数，それ以外の層は ReLu を用いる．また，損失関数には 2 値クロスエントロピーを用い，最適化関数には Adam [8]，学習率は 0.01，ドロップアウト層の不活性化確率は 0.2 とする．

3.3.2　水平結合の入力特徴量

水平結合の入力ベクトルである水平方向に隣接するトークン A(左) とトークン B(右) の特徴量を表 1，周辺の個々のトークンの特徴量を表 2 に示す．また，周辺のトークンの例を図 5 に

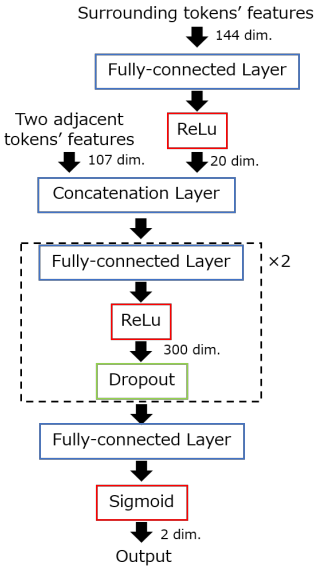

図 4　水平結合のモデル図

表 1　水平方向の隣接 2 トークンの特徴量

特徴	次元数
トークン A，B 間の距離	1
トークン A とトークン B のフォントの一致	1
トークン A とトークン B のスタイルの一致	1
トークン A のフォントサイズ	1
トークン B のフォントサイズ	1
トークン A のテキストは数値か	1
トークン B のテキストは数値か	1
結合位置	2
表のサイズ	2
トークンが属する列，行のトークン数	2
トークン A のテキストの品詞	47
トークン B のテキストの品詞	47
合計	107

示す．

まず，表 1 の水平に隣接する 2 トークンの特徴量について説明する．トークン A，B 間の距離は，トークン A の右端とトークン B の左端の差の絶対値である．フォントの一致やスタイルの一致は，両トークンについて一致していれば 1，そうでなければ 0 とする．トークンのテキストが数値かは，トークンのテキストが 0-9 の数字と，“.”，“-”，“%”，“$”，および数値に関連する単語である “greater”，“smaller”，“more”，“less” で構成されていれば 1，そうでなければ 0 とする．結合位置は，トークン A，B 間の中点の座標とし，表のサイズは表の幅と高さとする．トークンが属する列や行のトークン数は，そのトークンの上下の延長上にあるトークンの数を列のトークン数，左右の延長上にあるトークン数を行のトークン数とする．トークンのテキストの品詞は，Natural Language Toolkit(NLTK)[8] を用いて取得した 47 次元の one-hot ベクトルである．

次に，周辺のトークンの特徴量について述べる．トークン A，

8 : https://www.nltk.org

表 2　周辺の各トークンの特徴量

特徴	次元数
座標	2
幅	1
高さ	1
テキストか数値か	1
合計	5

	Recall	Precision	F-measure
Proposed method	0.95	0.86	0.90
Method A	0.91	0.84	0.87
Method B	0.87	0.86	0.86

図 5　周辺のトークンの例

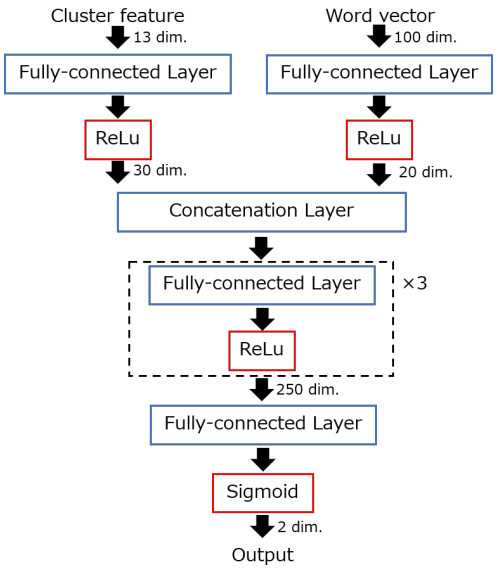

図 6　補助罫線推定のモデル図

B それぞれ自身と上下左右の隣接 4 トークン，重複も含め合わせて 10 トークンを周辺のトークンと定義する．図 5 で青の矩形で囲まれた箇所をトークン A，緑の矩形で囲まれた箇所をトークン B とすると，トークン A，B 自身と赤の矩形で囲まれた箇所が周辺のトークンとなる．表 2 の周辺の各トークンの特徴量は，トークンの座標，幅，高さ，テキストか数値かである．幅はトークンの左端の x 座標と右端の x 座標の差の絶対値であり，高さは上端の y 座標と下端の y 座標の差の絶対値である．周辺のトークンの特徴量は，上記の周辺の各トークンの 5 次元の特徴ベクトルと 94 次元の品詞タグを結合したベクトルである．よって，周辺のトークンの特徴量は 5×10+94 の 144 次元である．品詞タグには，トークン A の左に隣接する 47 次元のトークンのテキストの品詞の one-hot ベクトルとトークン B の右に隣接する 47 次元のトークンのテキストの品詞の one-hot ベクトルを連結したベクトルを用いる．

3.4　補助罫線推定

3.4.1　概　　要

本稿では，トークンの特徴とトークンのテキストの分散表現を用いてトークンの位置関係に基づく補助罫線を推定する．補助罫線推定モデルの概要を図 6 に示す．図 6 のモデルへの入力は，補助罫線候補の 13 次元の特徴ベクトル (Cluster feature) と補助罫線候補を構成する各トークンのテキストの 100 次元の分散表現の平均 (Word vector) である．また，出力は補助罫線か非補助罫線かの 2 次元である．中間層の出力次元数は，連結層より前では分散表現を入力とする層は 20，クラスタの特徴量を入力とする層は 30 とし，結合層より後では 250 とする．出力層の活性化関数は Sigmoid 関数，それ以外の層は ReLu を用いる．また，損失関数には 2 値クロスエントロピーを用い，最適化関数には Adam，学習率は 0.01 とする．

表 3　補助罫線推定へのクラスタの特徴量

特徴	次元数
クラスタを構成する点の数	1
表中の水平 (垂直) 方向のトークン数	1
補助罫線候補を挟むトークン間の罫線の有無	1
補助罫線候補の方向	1
クラスタの種類	6
補助罫線候補がセル上を通るか	1
補助罫線候補の位置	1
表のサイズ	1
合計	13

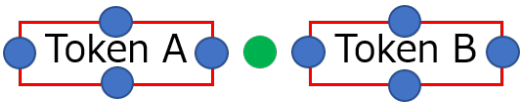

図 7　トークンの上下左右の端点と重心の中点

3.4.2　補助罫線推定の入力ベクトル

補助罫線推定の入力ベクトルのうちクラスタの特徴量を表 3 にまとめる．提案手法の補助罫線推定は，山田らのものに新たな入力ベクトルとして分散表現を追加したものになっている．補助罫線候補は，トークンの周辺に定める点をクラスタリングして得る．まず，垂直方向ではトークンの左端，右端，水平方向に隣接する 2 トークンの重心の中点，水平方向ではトークンの上端，下端，垂直方向に隣接する 2 トークンの重心の中点のそれぞれ 3 種類の点集合を作成する．そして，それぞれの集合において重心法でクラスタリングして得られた点集合のクラスタを補助罫線候補とする．トークンの上下左右の端点と重心の中点を，図 7 に示す．表 3 のクラスタにおける特徴量のクラスタの種類は，トークンの左端の x 座標，右端の x 座標，水平方向に隣接する 2 トークンの重心の中点の x 座標，上端の y 座標，下端の y 座標，垂直方向に隣接する重心の中点の y 座標で構成されたクラスタの 6 次元を表す one-hot ベクトルである．図 6 の Word vector はクラスタ中の各点の生成に関与したトークンのテキストの分散表現を平均したものである．この分散表現は Word2vec [9] により得ており，Word2vec の学習には Lipzig Copora[9]の English News(2016) と表を含む文章 PDF を pdfalto によりテキスト化したものを用いる．

9 : http://wortschatz.uni-leipzig.de/en/download/

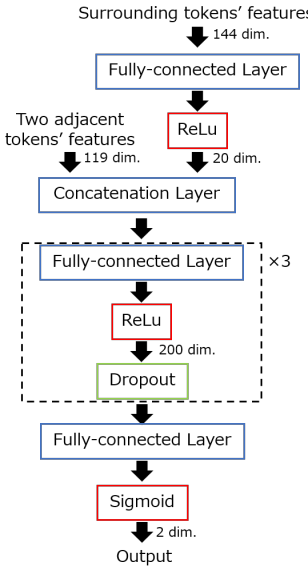

図 8　垂直結合とセル生成のモデル図

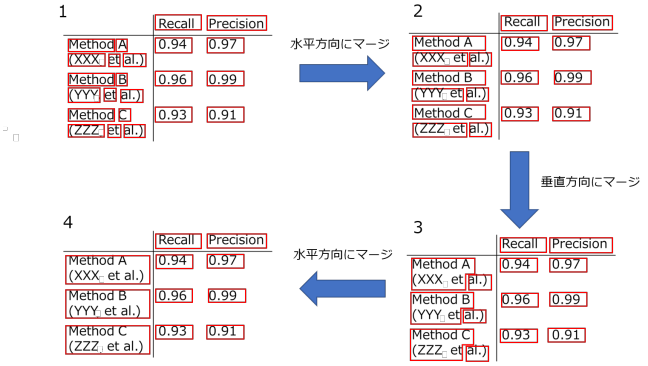

図 9　セル生成の例

3.5　垂直結合

3.5.1　概　　要

垂直結合は，垂直方向に隣接する 2 トークンの特徴と周辺のトークンの特徴を用いてその隣接 2 トークンを垂直方向にマージするものである．なお，マージする垂直方向に隣接する 2 トークンがなくなるまでマージをつづける．垂直結合のモデル図を図 8 に示す．なお，垂直結合と 3.6 節で説明するセル生成には図 8 の同じモデルを使用する．図 8 のモデルへの入力は，119 次元の垂直方向に隣接する 2 トークンの特徴ベクトル (Two adjacent tokens' features) と 144 次元の周辺のトークンの特徴ベクトル (Surrounding tokens' features) である．また，出力はマージするかしないかの 2 次元である．中間層の出力次元数は，連結層より前で周辺のトークンの特徴ベクトルを入力とする層は 20，結合層より後では 200 とする．出力層の活性化関数は Sigmoid 関数，それ以外の層は ReLu を用いる．また，損失関数には 2 値クロスエントロピーを用い，最適化関数には Adam，学習率は 0.01，ドロップアウト層の不活性化確率は 0.2 とする．

3.5.2　垂直結合の入力ベクトル

垂直結合の入力ベクトルのうち，隣接 2 トークンの特徴量を表 4 に示す．なお，周辺の各トークンの特徴量は，表 2 と同じものである．表 4 のトークン A とトークン B のテキストの一致度合いは，一致度合いが高い隣接 2 トークンは個別のセルとなる可能性が高いと考え導入したもので，python のライブラリである difflib を用いて算出する．周辺のトークンの特徴量では，トークン A の上に隣接するトークンのテキストの 47 次元の品詞の one-hot ベクトルとトークン B の下に隣接するトークンのテキストの 47 次元の品詞の one-hot ベクトルを連結したものをトークンの品詞として用いる．また，周辺のトークンの特徴量の次元は，3.3 節の水平結合の場合と同じく 5×10+94 の 144 次元である．

表 4　垂直結合とセル生成で利用する隣接 2 トークンの特徴量

特徴	次元数
トークン A，B 間の距離	1
トークン A とトークン B のフォントの一致	1
トークン A とトークン B のスタイルの一致	1
トークン A のフォントサイズ	1
トークン B のフォントサイズ	1
トークン A のテキストは数値か	1
トークン B のテキストは数値か	1
結合位置	2
表のサイズ	2
トークンが属する列，行のトークン数	2
結合の方向	2
間に存在するセパレータを構成する点の数	9
トークン A のテキストの品詞	47
トークン B のテキストの品詞	47
トークン A とトークン B のテキストの一致度合い	1
合計	119

3.6　セル生成

3.6.1　概　　要

セル生成では，水平結合と垂直結合がマージした後の隣接する 2 トークンの特徴と周辺のトークンの特徴を用いて，トークンをさらにマージすることによりセルを生成する．マージは水平方向と垂直方向に交互に行い，マージするトークンがなくなるまでつづける．

セル生成では，水平・垂直どちらのマージにも 3.5.1 項で述べたモデルと，3.5.2 項で述べた隣接 2 トークンの特徴量を用いる．提案手法のセル生成では，山田らの定めた特徴量に新たな特徴量としてトークンのテキストの品詞タグと隣接 2 トークンのテキストの一致度合いを追加し，隣接 2 トークンのテキストの分散表現を用いずにセルを生成する．

3.6.2　セルの生成過程

本項では，図 9 のセル生成の例を用いて説明する．図 9 の赤の矩形で囲まれた箇所はトークンである．セル生成は，まず，水平方向に隣接する 2 トークンをマージする．例えば図 9 では “XXX” と “et” がなどがまずマージされる．次に，垂直方向に隣接する 2 トークンをマージする．図 9 では，“Method A” と “XXX et” などがマージされている．そして，最後に水平方向に隣接する 2 トークンをマージする．図 9 では “method A

表 5　学習に用いた論文集とデータセット

論文集，データセット名	表数
NTCIR9 Spoken Doc [10]	27
NTCIR12 QA Lab-2 [11]	25
NTCIR12 SpokenQuery & Doc-2 [12]	16
Journal of Machine Learning Research Vol.18[10]	43
ICDAR2013 training dataset(EU)[11]	63
ICDAR2013 training dataset(US)[11]	35
合計	209

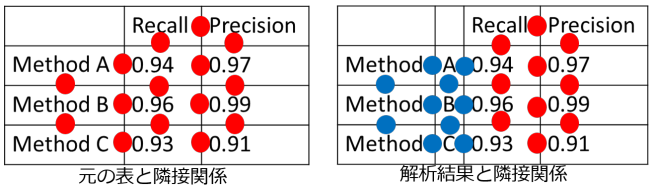

図 10　セルの隣接関係

XXX et”と “al.)”などがマージされている．

4　実　　験

4.1　データセット

本稿で提案する表構造解析の NN モデルの学習に用いる論文集およびデータセットを表 5 にまとめる．

表構造解析の評価には，ICDAR2013 Table competition のテスト用データセットを用いる．このデータセットは，EU，米国政府（US）の発行した様々なドメインの文書 PDF から表を収集したものであり，合計で 156 の表を含んでいる．

また，表画像を入力とする表構造解析実験の評価のためこの文書 PDF を ilovepdf.com[12]で画像に変換，得られた画像から同じ 156 の表の表画像を抽出した．なお，表画像の解像度は水平方向と垂直方向ともに 150dpi，ビットの深さは 24 とした．

4.2　評 価 指 標

評価指標には，Göbel らが定義した表中のセルの隣接関係に基づく評価指標 [13] と Zhong らが定義した Tree-Edit-Distance-based Similarity(TEDS) を用いる．

セルの隣接関係の例を図 10 に示す．図 10 の赤い点で示された箇所は正しい隣接関係，青い点で示された箇所は誤った隣接関係を表す．セルの隣接関係の再現率と適合率は，それぞれ (1) 式，(2) 式で算出される．また，F 値は再現率と適合率の調和平均である．

$$\text{再現率} = \frac{\text{解析結果の正しい隣接関係の数}}{\text{正解データの隣接関係の数}} \quad (1)$$

$$\text{適合率} = \frac{\text{解析結果の正しい隣接関係の数}}{\text{解析結果の隣接関係の数}} \quad (2)$$

一方 TEDS は，HTML 形式で表された正解データの表構造と解析結果の表構造の類似度を Tree-Edit Distance [14] に基づいて定めた類似度である．TEDS の算出式を (3) 式に示す．ここで T_a，T_b は正解データと解析結果の表，EditDist(T_a, T_b) は T_a, T_b の Tree-Edit Distance，$|T_a|$，$|T_b|$ はそれぞれの表のノード数である．

$$\text{TEDS}(T_a, T_b) = 1 - \frac{\text{EditDist}(T_a, T_b)}{\max(|T_a|, |T_b|)} \quad (3)$$

10 : http://www.jmlr.org/
11 : https://roundtrippdf.com/en/downloads/
12 : https://www.ilovepdf.com/ja/pdf_to_jpg/

表 6　表構造解析結果

手法名	再現率	適合率	F 値
提案手法	**0.967**	**0.977**	**0.972**
山田らの手法 [1]	0.951	0.960	0.955
Nurminen [15] (1st ranked)	0.941	0.952	0.946
2nd ranked [2]	0.640	0.614	0.627
3rd ranked [2]	0.481	0.570	0.522

表 7　提案手法の表構造解析の TEDS

入力	TEDS
PDF	**0.956**
表画像	0.831

表 8　水平結合のマージ結果

		推定	
		マージする	マージしない
正解	マージする	3,031	8
	マージしない	19	66

4.3　表構造解析実験

4.3.1　表構造解析精度

PDF を入力とした表構造解析の実験結果を表 6 に示す．提案手法は再現率が 0.967，適合率が 0.977，F 値が 0.972 となり山田らの手法をそれぞれ 1.6 ポイント，1.7 ポイント，1.7 ポイント上回った．

次に，提案手法で表の PDF を入力とした場合と表画像を入力とした場合の TEDS を計算した結果を表 7 に示す．なお，以降特に入力した表の種類の記述がない場合，PDF を入力とした表構造解析のことを指す．表 7 より，入力が PDF から表画像に変わると表構造解析精度が低下することが分かる．その要因として，表画像入力の場合，OCR で表中のトークンを認識するが，表中のトークンが検出できなかったり，文字認識を誤ったりすることが挙げられる．また本実験で使用した三つの NN は，PDF 入力で学習したものであることも要因の一つといえる．

4.3.2　水平結合の結果

実験における水平結合でマージすべきか判定される隣接 2 トークンは，マージするペアが 3,039 でありマージしないペアが 85 あった．マージするペアとマージしないペアの数が不均衡ではあるが，これは水平結合でマージすべきか判定するペアは，2 トークン間の距離が比較的近いペアを対象としているからである．

水平結合での隣接 2 トークンのマージ結果を表 8 に示す．

表 9　補助罫線推定の推定結果

		推定	
		補助罫線でない	補助罫線
正解	補助罫線でない	4,120	880
	補助罫線	568	12,564

表 10　セル生成のマージ結果

		推定	
		マージする	マージしない
正解	マージする	1,280	376
	マージしない	55	25,484

マージすべきなのにマージしなかったペアが 8，マージすべきでないのにマージしたペアが 19 あった．誤ってマージされた隣接 2 トークンと誤ってマージされなかった隣接 2 トークンが存在したが，マージすべき隣接 2 トークンの 99.7 %をマージすることができた．

4.3.3　補助罫線推定の結果

補助罫線推定の実験における補助罫線推定対象のクラスタ数は，補助罫線でないクラスタが 5,000 件，補助罫線であるクラスタが 13,132 件あった．補助罫線でないクラスタ数と補助罫線であるクラスタ数が不均衡であったため，SMOTEENN [16] で同数にしたものを学習に用いる．

補助罫線推定の推定結果を表 9 に示す．補助罫線でないクラスタを誤って補助罫線と推定する誤り (17.6%) が，その逆の誤り (4.3%) よりも割合として多かった．

4.3.4　セル生成の結果

セル生成の実験におけるマージすべきか判定される隣接 2 トークンは，マージするペアが 1,656 ありマージしないペアが 25,539 あった．マージするペアとマージしないペアの数が不均衡ではあるが，セル生成では誤ってマージされることが誤ってマージされないことよりも悪影響が大きいと考え，同数にはしない．

セル生成における隣接 2 トークンのマージ結果を表 10 に示す．マージすべきでないがマージした隣接 2 トークンが 55 あり，これは全体の 0.22 パーセントである．よって，マージすべきでない隣接 2 トークンに関してはほぼ正しく推定できているといえる．

5　考　　察

5.1　表構造解析結果の比較

山田らの手法の EU，US データセットの表構造解析結果 [1] を表 11 に，提案手法の EU，US データセットの表構造解析結果を表 12 に示す．表 11，表 12 より，提案手法は EU，US データセットのいずれのデータセットにおいても，山田らの手法の結果を上回ったことが分かる．特に，US データセットでは，F 値で 2 ポイント山田らの結果を上回った．また，US データセットは EU データセットよりも複雑な表構造をもつ表を含むデータセットであることから，提案手法は複雑な表構造をもつ表に対しても有効であると考えられる．

表 11　山田らの手法のデータセットごとの解析結果 [1]

	再現率	適合率	F 値
EU	0.987	0.984	0.986
US	0.939	0.952	0.945

表 12　提案手法のデータセットごとの解析結果

	再現率	適合率	F 値
EU	0.990	0.992	0.991
US	0.959	0.972	0.965

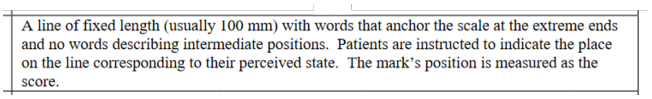
A line of fixed length (usually 100 mm) with words that anchor the scale at the extreme ends and no words describing intermediate positions. Patients are instructed to indicate the place on the line corresponding to their perceived state. The mark's position is measured as the score.

図 11　比較的長い文章を含むセル

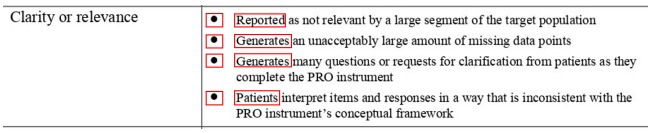
Clarity or relevance	• Reported as not relevant by a large segment of the target population • Generates an unacceptably large amount of missing data points • Generates many questions or requests for clarification from patients as they complete the PRO instrument • Patients interpret items and responses in a way that is inconsistent with the PRO instrument's conceptual framework

図 12　トークン間の距離が比較的大きいトークンのペア

図 11 に再現率，適合率が共に 1 であった，つまり解析に完全に成功した表のセルの一例を挙げる．図 11 のセルの中に比較的長い文章が含まれている．提案手法はまず水平結合で水平方向に隣接 2 トークンをマージすることで，垂直方向に誤ってマージされることがなくなったことが，解析に成功した要因として挙げられる．

次に，解析精度が悪かった表のセルの一例を図 12 に示す．図 12 のセルをもつ表の解析精度が悪かった要因として，セル中にトークン間の距離が大きいトークンが含まれていることが挙げられる．トークンのマージでは，トークン間の距離が大きいとマージされないことが多いが，図 12 で赤い矩形で囲まれた隣接 2 トークンは，マージされなかった．

5.2　トークンの水平結合の考察

誤って水平結合されたトークンの例を図 13 に示す．図 13 で赤い矩形で囲まれた二つのトークンが水平結合で誤ってマージされた．なお，“Less than”はその前に，トークン “Less”と “than”がマージされて生成されたトークンである．このトークンのペアが誤ってマージされた要因として，2 トークン間の距離が近いことや Less than $10,000- が一見意味をなすことが挙げられる．これを防ぐには，より広範囲のトークンの情報を考慮してトークンのマージを行う必要があることが考えられる．

5.3　補助罫線推定の考察

方向別の補助罫線推定の結果を表 13 に示す．表 13 から補助罫線については，垂直方向の補助罫線推定の結果が水平方向の補助罫線推定の結果より少し悪いことが分かる．これは，水平方向の補助罫線候補はそれが補助罫線でなかったとしてもそれを構成する点の数が多いことが要因として挙げられる．通常，補助罫線推定では，多くの点から構成される補助罫線候補が補助罫線と推定されやすい．その例を図 14 に示す．図 14 の表中

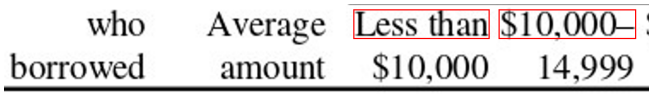

図 13　水平結合で誤ってマージされたトークン

表 13　方向別の補助罫線推定の結果

	再現率	適合率	F 値
垂直方向			
非補助罫線	0.921	0.861	0.890
補助罫線	0.905	0.947	0.926
水平方向			
非補助罫線	0.868	0.599	0.709
補助罫線	0.898	0.975	0.935

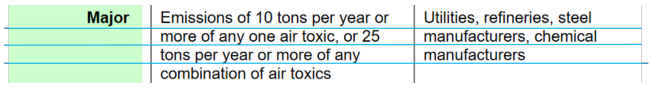

図 14　多くの点から構成される補助罫線でない補助罫線候補

の青い線分で表された補助罫線候補は，実際は補助罫線ではないが多くのトークンがこの補助罫線候補のクラスタ中の点を構成している．

6　おわりに

本稿では，ニューラルネットワークを用いた表構造解析手法を提案した．提案手法はまず，表中の水平方向に隣接する 2 トークンをマージする．次に，トークンの位置関係に基づき補助罫線を推定し，垂直方向に隣接する 2 トークンをマージする．最後に，水平方向に隣接した 2 トークンと垂直方向に隣接した 2 トークンを交互にマージすることでセルを生成する．

実験では，ICDAR 2013 の Table competition で提供されたテスト用データセットを用いて、文書 PDF を入力した場合と表画像を入力とした場合の表解析精度を評価した．その結果，PDF 入力の場合，提案手法はセルの隣接関係に基づく評価指標で再現率が 0.967，適合率が 0.977，F 値が 0.972 となった．この F 値は，ICDAR 2013 Table competition の参加者の最高結果を上回った山田らの手法の F 値を 1.7 ポイント上回った．また，入力を PDF から表画像に変更すると，表構造解析精度を示す指標の TEDS が 0.956 から 0.831 に低下した．これは，表中のトークンの OCR による認識誤りや，表画像解析に用いた NN が PDF 入力で学習したモデルであることが要因である．

今後の課題としては，表構造解析結果を用いてグラフを生成するアプリケーションの開発等が挙げられる．

文　献

[1] 山田凌也，太田学，金澤輝一，高須淳宏. 機械学習を用いた表構造解析の一手法. 第 12 回データ工学と情報マネジメントに関するフォーラム，E6-4，2020.

[2] M. Göbel et al. ICDAR 2013 table competition. In Proceedings of the 12th International Conference on Document Analysisand Recognition, pp. 1449-1453, 2013.

[3] S. R. Qasim, H. Mahmood, and F. Shafait. Rethinking table recognition using graph neural networks. In Proceedings of 2019 International Conference on Document Analysis and Recognition, pp. 142-147, 2019.

[4] S. Paliwal, D. Vishwanath, R. Rahul, M. Sharma, and L. Vig. TableNet: Deep learning model for end-to-end table detection and tabular data extraction from scanned document images. In Proceedings of 2019 International Conference on Document Analysis and Recognition, pp.128-133, 2019.

[5] S. Schreiber, S. Agne, I. Wolf, A. Dengel, and S. Ahmed. Deepdesrt:Deep learning for etection and structure recognition of tables in document images. in Document Analysis and Recognition(ICDAR), 2017 14th IAPR International Conference on, vol. 1. IEEE, 2017, pp.1162-1167.

[6] X. Zhong, E. ShafieiBavani, and A. J. Yepes. Image-based table recognition: data, model, and evaluation. 16th European Conference on Computer Vison, 2020.

[7] Y. Deng, A. Kanervisto, J. Ling, and A. M. Rush. Image-to-markup generation with coarse-to-fine attention. In Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR.org, pp. 980-989, 2017.

[8] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In Proceedings of International Conference on Learning Representations(ICLR), 2015.

[9] T. Mikolov et al. Efficient estimation of word representations in vector space. CoRR. vol. abs/1301.3781, 2013.

[10] A. Tomoyosi et al. Overview of the IR for spoken documents task in NTCIR-9 workshop. In Proceedings of the 9th NTCIR Workshop Meeting, pp. 223-235, 2011.

[11] H. Shibuki et al. Overview of the NTCIR-12 QA Lab-2task. In Proceedings of the 12th NTCIR Workshop Meeting, pp. 392-708, 2016.

[12] T. Akiba et al. Overview of the NTCIR-12 SpokenQuer-y & Doc-2 task. In Proceedings of the 12th NTCIR Workshop Meeting, pp. 167-179, 2016.

[13] M. Göbel, E. Oro, and G. Orsi. A methodology for evaluating algorithms for table understanding in PDF documents. In Proceedings of the ACM Symposium on Document Engineering 2012, pp. 45-48, 2012.

[14] M. Pawlik and N. Augsten. Tree edit distance: Robust and memory-efficient. Information Systems, vol. 56, pp. 157-173, 2016.

[15] A. Nurminen. Algorithmic extraction of data in tables in pdf documents. Master's thesis, Tampere University of Technology, 2013.

[16] G. Batista, R. Prati, and M. Monard. A study of the behavior of several methods for balancing machine learning training data. SIGKDD Explorations, vol. 6, pp. 20-29, 2004.