Vol.2016-NL-226 No.16
Vol.2016-SLP-111 No.16
2016/5/17

表形式のトピックモデルとその数値単位推定への応用

吉田 稔[1,a)]　松本 和幸[1,b)]　北 研二[1,c)]

概要：表形式中の数値表現について単位が省略されている場合に，その単位を推定するための手法を提案する．Wikipedia 表形式中の，数値のみを含むセルを対象に，一行目のセル等の「周辺文脈」を利用し，適切な単位を推定する．また，表形式の外側の文章を利用するために，表形式と文章を同時にモデル化するためのトピックモデルを提案し，トピックの推定結果を単位推定に利用することで精度の向上を図る．

1. はじめに

数値は，テキスト中の知識，特に，科学的・客観的記述を抽出するための重要な情報源であるが，テキスト中の数値情報から知識を抽出する研究は，主に情報抽出や質問応答システムの中のモジュールとして，単に与えられた属性の属性値を抽出する処理の一環として扱われることが多く，複数の数値記述をまとめて知識として抽出する研究は少ない．

一方，表形式は，数値データを表現する際に一般的に用いられる表現手段である．表形式は，類似するエンティティ同士を区別するのに重要な特徴を「属性」とし，各エンティティの属性値を列挙することで，簡潔に記述することができる．

テキスト中，特に，表形式中の数値を扱う際問題となるのが，数値のうちの多くについて，単位が明らかな場合，それが省略されるということである．特に，表形式では，属性名や，他の属性値などの周辺情報が多く記述されているため，人間にとって単位が明らかな場合が多い．このため，多くの場合に単位が省略される．（例えば，ランキング形式の表の場合，「順位」という属性の属性値では，しばしば単位「位」が省略される．）このように省略された単位を，周辺情報から推定するタスクを考える．このような単位推定を行うことにより，例えば，表形式中のデータを，(数値, 単位) のペアにより検索・マイニングする，等の応用に役立つことが考えられる．

本研究の貢献は，主に 3 つである．まず，「数値の単位推定」という新たなタスクを，特に表形式に注目し提案すること．次に，テキスト中の数値表現に対し，「有効数字」の概念を用いた効果的なモデル化手法を提案すること．最後に，文書中の表形式と周辺テキストを同時にモデル化する Table Topic Model の提案である．

[1] 徳島大学大学院理工学研究部
Tokushima University
[a)] mino@is.tokushima-u.ac.jp
[b)] matumoto@is.tokushima-u.ac.jp
[c)] kita@is.tokushima-u.ac.jp

2. 関連研究

表形式中の単位推定を扱った論文は，著者らの知るかぎりでは存在しないが，テキスト中の数値情報，表形式情報を扱った研究に関しては多くの既存研究が存在する．

2.1 テキスト中の数値情報

テキストと数値情報に関する研究としては，例えばメタデータ中の数値情報を利用したインデックス付け [5] によるメタデータによる効率的な絞り込み検索の研究や，Google による数値範囲（“100..200 dollars” 等）による検索 “search by numbers” 等，Web 検索を提供する企業によるものが存在する．また，数値範囲クエリを可能としたテキストマイニングシステム [16] や，数値に関する「常識的範囲」を推定するシステム [9][14] 等も提案されている．しかしながら，これらは，表形式の取り扱いについては考慮していない．

2.2 表形式

表形式を知識源として用いる研究も多く存在し，表形式中の属性共起に関するマイニング [3]，複数クエリ・複数列に対応した表形式検索 [12]，表形式の各セルを，知識ベースへの対応付けする研究 [7]，Wikipedia 表形式からの三つ組抽出 [8] 等の研究がある．また，Sarawagi ら [13] は，Web 中の表形式の数値クエリによる検索を提案している．彼らの手法では，表形式に明示的に示された数値と単位を対象としているため，我々の単位推定タスクとは補完的関係にある．

表形式と周辺テキストを関連付ける研究として，Govindaraju ら [6] は，情報抽出のために，テキストおよび表形式から抽出された素性を統合して利用する手法を提案している．また，Wang ら [15] は，表形式が与えられた時に，その周辺の文を表形式に関連しているかどうか分類する手法を提案している．これに対し，本研究の提案である表形式の意味解析，特に周辺の文まで含めた解析を，トピックモデルを用いて行った研究は，筆者らの知る限りではこれまで存在しない．

また，表形式の属性位置推定に関しては，多くの既存研究がある．[4][17]．しかしながら，属性位置推定は間違いの混入が避けられないため，本研究では，推定を行わず，可能性のあるセルをすべて文脈として用いる方針を取る．

3. 問題設定および使用データ

データとして，Wikipedia 日本語版（2013 年ダウンロード）を利用する．このデータに対し，数値のみを含むセルを，単位が省略されたセルと見なし，適切な単位を推測するタスクを設定した [*1]．

3.1 表形式のセルへの分解

タスク設定に際し，表形式を，セルの集まりに分解する [*2]．この設定において，システムへの入力は，ペア (x_i, y_i) のリストであり，ここで，データ x_i は各セルを表し，数値とその周辺（文脈）単語の列となる．（単語列は，各次元を各単語に対応させたベクトルに変換して扱う．）また，ラベル y_i は，数値に対応する単位である．本論文での問題設定は，ラベル y_i が未知のときに，その値を推測するというものである．

使用したデータ中，テーブル数は 255,039 であり，そのうち 78,967 個はテンプレート参照を含むため除外し，残る 176,072 個を対象データとした．

我々の扱う Wikipedia のテキストから，数値を含むセル 59,847 個をサンプリングした結果，数値以外の単語数が 1 つ以下のものは 29,397 個（49.1%）であり，さらに，そのうち，23,423（39.1%）が数値のみのセルであった．すなわち，約 4 割の数値セルにおいて，単位が省略されていると見積もることができる．このうち，単位のないセルをランダムに 297 個取得し，人手により単位を付与した．

[*1] 単位を当てはめることができないものについては，NULL という単位を割り当てる．

[*2] このとき，文脈は各セルごとにコピーされることになる．我々は，表形式とは，同一の文脈を持つ様々な値を簡潔に表現するための手段であると仮定し，表形式を構築する際に，そのような文脈が統合されて省略されたものだと仮定する．（例えば，「年齢：25 歳」「年齢：30 歳」という 2 つの表現が統一され，属性名として「年齢」が一回のみ出現する表形式に変換されると考える．）この意味で，タプルへの変換は，このように省略された文脈を復元する操作と考えることができる．

3.2 単位推定のための文脈情報

例えば，単位が「円」のとき，それらの文脈語は，「価格」等の単語を含む可能性が高い．また，数値そのものも，単位推定の手掛かりとなりうる．例えば，「1987」という数値からは，単位「年」が当てはまる可能性が高いため，単位毎，あるいはトピック毎の数値の分布をモデル化することが望ましい．提案手法では，数値セルに関連が深いと思われる位置（例えば，セルと同列の 1 行目のセル）の単語情報を，文脈情報として単位推定のために用いる．具体的には，以下の情報を用いる．

数値： 数値セル内の数値そのもの．

第一行・第一列セル： 表形式の中には，再帰的構造や，2 行・2 列以上に渡る属性セル等，複雑な構造を持つものもあるが，多くの場合，特に Wikipedia 中の表形式に関しては，最初の行（または最初の列）に属性が表示されるという単純な構造を持つ．本研究では，すべての表形式にこれらの基本的な構造を仮定し，これに基づき，「同列 1 行目」と「同行 1 列目」の 2 つのセルを文脈として用いる．なお，例えば属性が第一行目にある場合，多くの場合で，第一列目にはエンティティの名称等，重要な情報が記載されているため，行・列どちらか一方のみを選ぶことはせず，両方を文脈として利用する．

見出し： 表形式中に属性名が表記されていない場合でも，見出し中に属性名が表記されている場合が少なからず存在する．当該表形式に関連しそうな文書中の文字列として，文書の階層的構造（文書構造）に着目する．Wikipedia の文書は，見出しを定義するルールが定められているため，これに従い，表形式を含む部分文書の見出し（表形式直前の見出し）を取得し，これを文脈単語とする．

本論文では，これら文脈を w_i, その集合を C と表記する．これらを基本素性と呼ぶが，これに加えて，実験では，後述の通り，同一列・同一行の他のセル中に出現する単語および数値の情報を素性として追加した場合も考慮する．

4. 単位推定手法

4.1 Logistic Regression による単位推定

単位推定のために，(regularized) logistic regression [1] を行う．このとき，目的関数は

$$f(\lambda) = -l(\lambda) + r(\lambda)$$

（ここで，$l(\lambda)$ は入力データの対数尤度，$r(\lambda)$ は正則化項．パラメータ λ は，入力データ x_i と同一の次元数を持つ．）であり，これを最小化する λ を求めるのが目的となる．

v 個目のデータとして x を観測する確率を $\pi(x)_v$ を，

$$\pi(x)_v = \frac{e^{\lambda_v \cdot x}}{\sum_{u=1}^{k} e^{\lambda_u \cdot x}}$$

とすると，尤度関数は

$$\prod_{i=1}^{m} \pi(x(i))_{y(i)}$$

と定義され，正則化項は，

$$C \sum_i |\lambda_i|^p$$

となる．ここで，p は正則化のためのパラメータであり，$p = 1$ ならば L1-正則化，$p = 2$ ならば L2-正則化となる．C は正則化項の重みを決めるパラメータである．

4.2 提案手法: Table Topic Model

Table Topic Model は，LDA を，表形式も含んでモデル化するように拡張したモデルである．

Latent Dirichlet Allocation (LDA) [2] では，文書中の各単語に，

(1) パラメータ θ_d を，Dirichlet 分布に従って選ぶ：$\theta \sim Dir(\alpha)$

(2) 各単語について，

(a) トピック z を，多項分布に従って選ぶ：$z \sim Multi(\theta_d)$

(b) 単語 w を，トピック z の単語分布に従って選ぶ：$p(w|z)$

という生成過程を仮定する．ここで，d は，現在注目している文書の id を表す．トピック z は，θ を積分消去した後，z に関する Gibbs Sampling を行うことによって推定されることが多い．このとき，Gibbs Sampling は，推定する単語以外のトピック配置から，推定する単語のトピックの確率を求めた，

$$P(w_i|z_i = j, \mathbf{z_{-i}}, \mathbf{w_{-i}}) = \frac{n^{w_i}_{-i,j} + \beta}{n^{\cdot}_{-i,j} + W\beta}$$

および

$$P(z_i = j|\mathbf{z_{-i}}) = \frac{n^{d_i}_{-i,j} + \alpha}{n^{(d_i)}_{-i,\cdot} + K\alpha}$$

に基づき行われる．ここで，$\mathbf{z_{-i}}$ は，i 番目の単語以外のトピック配置，$n^{\cdot}_{-i,j}$ は，現在の単語を除くすべての単語のうち，トピック j に属する単語の数である．

LDA では，文書の各単語が，別々のトピックから生成されることを仮定している．同様に，表形式での各セルを，別々のトピックから生成されるとして混合モデルを構築することは可能であるが，この場合，「同一列，あるいは同一行のセル同士が関連付けられている」という表形式の重要な性質を活用できない．

そこで本研究では，

- 表形式の各列について，単一のトピックから生成されることを仮定し，
- 表形式のトピックは，周辺の文章のトピックと共有される．

という仮定を置くことにより，

- 文章と表形式の自然な関連付け
- 表形式中のセル同士の自然な関連付け

を同時に行える新たなトピックモデル（Table Topic Model）を提案する．

Table Topic Model では，表形式の同一列の単語が同一のトピックに属することを仮定し，表形式の各列に 1 つのトピックを割り当てる．我々のデータセットでは，9 割以上の表形式が row-wise（属性が横に並ぶ表）であり，Wikipedia 中の表形式については，このモデルがある程度妥当性を持っていると考えられる．ここで，表形式中の各列を $C =< c_1, c_2, \cdots, c_{|C|} >$ なる単語列とすると，各列に対し，

(1) トピック z を，多項分布に従って選ぶ：$z \sim Multi(\theta_d)$

(2) 各単語について，

(a) 単語 c_i を，トピック z の単語分布に従って選ぶ：$p(c_i|z)$

という生成過程を仮定したユニグラム混合分布 (Mixture of Unigrams, Multinomial Mixtures) としてモデル化する．ここで，表形式のためのトピック分布と，本文中のトピック分布は共有される．これにより，例えば表形式中の属性名や単位名，後述の数値そのもの等を手がかりに，表形式中の各列について，「その列とトピックを共有する本文中の単語の集まり」が得られることになり，各セルの文脈取得の範囲を拡張することができると考えられる．

トピック推定の際は，各列に対し，各トピックを割り当てた際の確率値をそれぞれ求め，それに基づいて Gibbs Sampling を行う．

4.3 数値のモデル

本研究では，数値（本文中，表形式中とも）に関しては，他の単語とは異なるモデル化を行う．ここでは，実数等の連続値に対する事前確率を定義するための確率モデルである Polya Tree[10] にヒントを得た，各桁の数値を段階的に生成するモデルを用いる．また，テキスト中の数値が，どのような文字列で記述されているか，という「記述形式」自体が，その数値が人間にとってどのような意味付けかを表現しているという考えから，コード化の際にも，「どの桁までが記述されているか」の情報を組み込む．数値を離散的にモデル化することで，単語の分布と数値の分布を自然に統合できる．

提案モデルでは，数値の生成モデルを構築する際，その「有効数字」を利用する．数値 d は，以下の手順により，連続する数字列 e_i（以下，コード）に変換される．

(1) e_1：符号 ($d \geq 0$ のとき 1)
(2) e_2, e_3：d の最大桁（e_2 は桁数の符号，e_3 は絶対値を表現する．すなわち，$e_3 = log_{10}(d)$)
(3) e_4：0 でない桁の長さ (記述者が重要だと考える桁数を表現)
(4) e_5 以降は，実際の有効数字を記述する [*3].

これにより，例えば，「-9.5」は，<0，2，1，3，9，5>. と変換される．

4.4 生成モデルのための変換

上述のコードへの変換を行った後，変換後の数値列に確率値を与える．本研究では，各数字が多項分布に従って出力され，その多項分布は Dirichlet 分布から出力されると仮定する．

(1) 各数字 e_i は，$H(e_1 \ldots e_{i-1})$ に従って選ばれる: $e_i \sim H(e_1 \ldots e_{i-1})$.
(2) 各 H_i は，Dirichlet 分布に従って選ばれる: $H_i \sim Dir(H)$.

ここで，$H(e_1 \ldots e_{i-1})$ は，各数字列 $e_1 \ldots e_{i-1}$ 毎に定義される多項分布である．Dirichlet 分布のパラメータ H は，一様分布を仮定する．（すなわち，すべての数字を当確率とする．）e_i の分布は，各 H_i を積分消去することで，以下のようにして得られる．

$$P(e_i) = \frac{n_{ec}}{\alpha + n_{.c}} + \frac{\alpha}{\alpha + n_{.c}} H(e_i) \quad (1)$$

ここで，n_{ec} は e の文脈 $c = (e_1 \ldots e_{i-1})$ における出現回数，$n_{.c}$ は，文脈 c におけるすべての数字の出現回数の和である．

4.5 識別モデルのための変換

テキスト中の数値情報を識別モデルで利用する際も，前述のコード化を利用する．利用方法として，本研究では，以下の 3 つの手法を検討する．

i.i.d. coding コード中の各数字を，独立に素性として扱う．例えば，「-9.5」は<0,2,1,3,9,5>に変換され，それぞれの数字を，例えば N00, N12, N21, N33, N49, N55 のような単語として扱う．（ここで，頭文字 N は，数値素性であることを表す．2 つ目の数字はコード中の位置，3 つ目の数字は数字そのもの．）

Polya coding i.i.d. コーディングと異なり，各数字について，その左側のすべての数字も，履歴として同時にコード化する．例えば，「-9.5」は，<0,2,1,3,9,5>とコード化された後，それぞれ N0, N02, N021, N0213, N02139, N021395 という単語として扱われる．

quantization 一つのコードを，一つの単語に変換する．このさい，最も詳細化されたコードを用いる．例えば，「-9.5」は<0,2,1,3,9,5>と変換されたあと，すべての数字を用いた N021395 という単語として扱われる．これは，「有効数字」で抽象化された数値をそのまま用いることに相当する．

[*3] 現在は，2 桁分を使用．

5. 実験

3 節で説明したデータに対し，単位推定の実験を行った．単位のうち，頻度が 2 以上の単位のみ（41 種）を使用する．この結果，297 セルのうち，270 セルを実際に評価に用いた．5 分割交差検定による各手法の正解率を測定する．交差検定の際には，同一の表形式からのセルが訓練とテストに跨がらないような分割を行った．

また，識別モデルにおける数値の素性への変換に関して，以下のベースラインも加える．

nonum: 数値に関する素性を用いない．
raw: 数値を表す文字列をそのまま素性として用いる．
double: 数値の値を解釈した double 値を表す文字列を素性として用いる．

実験では，3 節で述べた基本素性のほか，

column: 評価中のセルと同一の列にあるセルを全て文脈として加える．
column and row: 評価中のセルと同一の列あるいは同一の行にある全てのセルを文脈として加える．

の 2 種類の手法で素性を拡張した場合の精度についても測定を行う [*4]．これは，Table Topic Model が同一列の情報をトピック推定に用いているため，利用情報を揃えるための設定である．

Logistic 回帰には，Classias [11] を用いた．Logistic 回帰のパラメータは，予備実験で最も高い性能を示した設定として，L1 正則化とパラメータ $C = 0.1$ を使用した．

Wikipedia の記事のうち，表形式を 1 つ以上含む記事を対象とし，提案モデルによるトピック推定（各単語，および表形式の各列）を行った．Table Topic Model のトピック数に関しては，k=10（トピック数 10）が最も良い精度を示したため，これを用いる．パラメータ推定は，Gibbs Sampling により行う．反復回数を 500 回とし，最後の 200 回でトピックをサンプリングし，各トピックの出現回数からトピックの分布を計算し，1 回以上出現したトピックに関して，トピック番号を素性に加える．そのさい，そのトピックの出現割合を素性の重みとする．Gibbs Sampling を用いた結果は，5 回の試行の平均を計算した．

6. 結果と考察

表 1 に結果を示す．トピック利用の有無にかかわらず，同一列の文脈を用いることにより（column および column and row），概ね精度が向上した．

[*4] なお，同一行の文脈のみを利用した場合についても実験を行ったが，精度は大きく低下したため，ここでは割愛する．

トピックを利用した場合の精度は，一部の設定 *5 を除き，利用しない場合よりも高くなった．トピックを利用することが，文書内の，同一内容の列どうしをまとめるクラスタリングの効果を持ち，利用できる文脈の範囲が広がったためであると考えられる．特に，識別モデルにおいて数値素性を用いなかった場合（nonum），トピックを素性に加える事の効果が大きかった．トピック推定の際に数値の情報を用いているため，トピック番号が，数値の情報を簡潔に表現できていることが理由であると考えられる．また，数値文字列をそのまま用いた場合（raw）は，トピックを用いない場合の精度は低かったが，トピックを用いることで最高精度を達成している．トピック分布が Polya モデルを用いて推定されているのに対し，raw モデルは文字列をそのまま用いる対照的なモデル化手法であるため，相互に素性情報が補完される効果があったと推測される．

また，トピック推定の際，表形式のみを利用した場合（table only）と，周辺の文章まで利用した場合（table and sentence）での比較を行ったところ，明確な精度差は観測できなかったものの，table and sentence の設定 *6 で最高精度を達成したほか，パラメータ 18 種のうち 13 種で table only の精度を上回ったことから，文章を利用することで，より質の高いトピック推定が行えていると推測できる．

7. おわりに

文書中の表形式と，本文中の単語を同時にモデル化する新しいトピックモデルを提案した．提案手法は，本文中と表形式中でトピック分布を共有することで，表形式の各セルの意味付けを，本文中の豊富なテキストを利用して行うことを目指した．また，数値を有効数字の考え方を利用して生成モデルに統合する手法を提案した．評価実験では，提案手法によるセルの意味付けが，単位推定タスクの精度を向上させ，文脈情報の取得に一定の効果を発揮することを確認した．今回のタスクでは，Wikipedia の表形式を対象としたが，適用範囲を一般の Web 文書に拡張することが今後の課題である．Wikipedia の表形式では，一行目に属性が表示されることが多かったが，一般の Web 文書を対象とする場合は，多様な構造を持つ表形式に対応するため，例えば，表の構造推定を，確率モデルに組み込む等の研究が必要となると考えられる．また，本手法を用いて文章中の数値に意味付けを行い，構文解析等のより深い文処理の結果と組み合わせることで，文中の数値の意味付けをより精緻に行うことも，今後の重要な研究課題である．

表 1　単位推定精度の比較

数値表現	基本素性	column	column and row
TOPIC なし			
nonum	67.41	69.63	65.93
i.i.d.	69.63	70.74	68.52
polya	71.85	72.96	72.22
quantize	70.37	71.11	72.22
raw	65.56	69.63	66.67
double	65.56	69.63	67.04
TOPIC あり（table only）			
nonum	70.22	73.11	73.04
i.i.d.	71.70	70.37	69.56
polya	72.44	72.74	72.96
quantize	72.07	74.44	73.41
raw	71.33	73.63	72.74
double	71.33	73.63	72.81
TOPIC あり（table and sentence）			
nonum	71.48	73.48	73.26
i.i.d.	71.93	70.00	69.33
polya	72.52	72.67	72.67
quantize	71.78	75.04	73.26
raw	72.44	**75.48**	73.48
double	72.44	**75.48**	73.33

*5 (i.i.d, column) および (i.i.d, Polya)
*6 (raw, column) および (double, column) の設定において．

謝辞 本研究は JSPS 科研費 15K00425, 15K00309, 15K16077 の助成を受けたものです．

参考文献

[1] Andrew, G. and Gao, J.: Scalable training of L1-regularized log-linear models, *Proceedings of ICML 2007*, pp. 33–40 (2007).
[2] Blei, D. M., Ng, A. Y. and Jordan, M. I.: Latent Dirichlet Allocation, *Journal of Machine Learning Research*, Vol. 3, pp. 993–1022 (2003).
[3] Cafarella, M. J., Halevy, A. Y., Wang, D. Z., Wu, E. and Zhang, Y.: WebTables: exploring the power of tables on the web, *Proceedings of VLDB Endowment 1(1)*, pp. 538–549 (2008).
[4] Embley, D., Hurst, M., Lopresti, D. and Nagy, G.: Table-processing paradigms: a research survey, *International Journal on Document Analysis and Recognition*, Vol. 8(2), pp. 66–86 (2006).
[5] Fontoura, M., Lempel, R., Qi, R. and Zien, J. Y.: Inverted Index Support for Numeric Search, *Internet Mathematics*, Vol. 3(2), pp. 153–186 (2006).
[6] Govindaraju, V., Zhang, C. and Re, C.: Understanding Tables in Context Using Standard NLP Toolkits, *Proceedings of ACL2013* (2013).
[7] Limaye, G., Sarawagi, S. and Chakrabarti, S.: Annotating and Searching Web Tables Using Entities, Types and Relationships, *Proceedings of VLDB Endowment 3(1)*, pp. 1338–1347 (2010).
[8] Munoz, E., Hogan, A. and Mileo, A.: Triplifying Wikipedia's Tables, *Proceedings of the ISWC 2013 Workshop on Linked Data for Information Extraction* (2013).
[9] Narisawa, K., Watanabe, Y., Mizuno, J., Okazaki, N. and Inui, K.: Is a 204 cm Man Tall or Small ? Acquisition of Numerical Common Sense from the Web, *Proceedings of the ACL (1)*, pp. 382–391 (2013).
[10] Neath, A. A.: Polya tree distributions for statistical modeling of censored data, *Journal of Applied Mathematics and Decision Sciences*, Vol. 3(7), pp. 175–186 (2003).

[11] Okazaki, N.: Classias: A collection of machine-learning algorithms for classification.
[12] Rakesh Pimplikar, S. S.: Answering Table Queries on the Web using Column Keywords., *Proceedings of VLDB Endowment 5(10)*, pp. 908–919 (2012).
[13] Sarawagi, S. and Chakrabarti, S.: Open-domain quantity queries on web tables: annotation, response, and consensus models, *Proceedings of KDD*, pp. 711–720 (2014).
[14] Takamura, H. and Tsujii, J.: Estimating numerical attributes by bringing together fragmentary clues, *Proceedings of NAACL-HLT2015* (2015).
[15] Wang, H., Liu, A., Wang, J., Ziebart, B. D., Yu, C. T. and Shen, W.: Context Retrieval for Web Tables, *Proceedings of ICTIR 2015*, pp. 251–260 (2015).
[16] Yoshida, M., Sato, I., Nakagawa, H. and Terada, A.: Mining Numbers in Text Using Suffix Arrays and Clustering Based on Dirichlet Process Mixture Models, *Proceedings of the PAKDD (2)*, pp. 230–237 (2010).
[17] Zanibbi, R., Blostein, D. and Cordy, J. R.: A survey of table recognition, *International Journal on Document Analysis and Recognition*, Vol. 7(1), pp. 1–16 (2004).