一般口演 | データベース・データウェアハウス
一般口演16
データベース・サーバ構築
2021年11月20日(土) 16:30 〜 17:30 D会場 (2号館1階212)

[3-D-2-03] PDF形式で保存される検査レポートからデータの抽出と構造化を実現するプログラムの開発

*区 駿業[1]、和田 聖哉[2,1]、武田 理宏[1]、岡田 佳築[2,1]、真鍋 史朗[1]、小西 正三[1]、杉本 賢人、松村 泰志[3,1]（1. 大阪大学大学院医学系研究科　医療情報学, 2. 大阪大学大学院医学系研究科　変革的医療情報システム開発学寄附講座, 3. 国立病院機構大阪医療センター）
*Junye Ou[1], Shoya Wada[2,1], Toshihiro Takeda[1], Katsuki Okada[2,1], Shirou Manabe[1], Shozo Konishi[1], Kento Sugimoto, Yasushi Matsumura[3,1]（1. 大阪大学大学院医学系研究科　医療情報学, 2. 大阪大学大学院医学系研究科　変革的医療情報システム開発学寄附講座, 3. 国立病院機構大阪医療センター）
キーワード：PDF, Acquired, Data Accuracy, Data Format

背景：電子カルテデータの二次利用に際して、後ろ向き研究では、非構造化データの取得が課題となる。検査レポートは検査種ごとに特徴的な書式で出力されるため、非構造化データではあるが構造化を目指した情報抽出ができる可能性が高い。 目的： PDF形式の検査レポートから、臨床研究に必要なデータを取得して構造化するプログラムを作成し、その抽出精度を検証する。 方法：大阪大学医学部附属病院で作成された放射線検査レポートと呼吸機能検査レポートから、それぞれ100枚、125枚を学術研究目的で抽出した。そのうち各10枚を評価用データとして、それら以外を抽出ルール作成のための学習データとして使用した。プログラミング言語は JavaとJavaScriptを使用した。 PDFにはテキストとそれに対応する座標が保存されているが、内部に格納されている順番が必ずしも画面上で表現されるような順番になっているとは限らない。抽出ルール構築のために、 PDF内の目的データを Key-Value型とテーブル型の二つに分類した。抽出ルールを学習データから構築した後に、簡単なHTMLインターフェースを使って、データの座標エリアを特定した。分類とその対応するエリアに合致するルールを適用してデータ抽出と構造化を行い、 JSON形式で保存した。最後に、構築した抽出ルールの精度を評価用データで検証した。 結果：本プログラムを用い、 Key-Value型データについては、全て正確に取得し、 JSONにフォーマットすることが出来た。テーブル型データの項目名と対応する値に誤りがあり、その抽出精度は78.7%であった。 結語：電子カルテに蓄積された PDF形式の検査レポートから高精度にデータを抽出して構造化することが可能であることを確認した。

©一般社団法人 日本医療情報学会

PDF 形式で保存される検査レポートからデータの抽出と構造化を実現するプログラムの開発

区 駿業[*1]、和田 聖哉[*2 *1]、武田 理宏[*1]、岡田 佳築[*2 *1]
真鍋 史朗[*1]、小西 正三[*1]、杉本 賢人[*1]、松村 泰志[*3 *1]
*1 大阪大学大学院医学系研究科 医療情報学,
*2 大阪大学大学院医学系研究科 変革的医療情報システム開発学寄附講座,
*3 国立病院機構大阪医療センター

Development of a program that realizes data extraction and structuring from inspection reports saved in PDF format

Junye Ou[*1], Shoya Wada[*2, *1], Toshihiro Takeda[*1], Katsuki Okada[*2, *1],
Shirou Manabe[*1], Shozo Konishi[*1], Kento Sugimoto[*1], Yasushi Matsumura[*3, *1]
*1 Osaka University Graduate School of Medicine Department, Medical Informatics,
*2 Department of Transformative System for Medical Information, Osaka University Graduate School of Medicine,
*3 National Hospital Organization Osaka National Hospital

In the secondary use of electronic medical record data for retrospective studies, unstructured data acquisition becomes an issue. Inspection reports are often distinct layouts for each type of examination. We developed a program to obtain and format the inspection results necessary for clinical research from the inspection reports saved in PDF format and verify their accuracy. We used the programming languages Java and JavaScript to develop the program. In the program, we read the contents of PDF files into strings by utilizing PDF layout characters and coordinate information and then classify the data into key-value-format or table-format. We obtain the coordinate range of the corresponding data through a simple HTML interface, use different rules to extract and format each kind of data. We experimented with acquiring the MRI reports and respiratory examination reports saved in PDF format using this program. As a result, it is possible to acquire and format the result values from PDF documents stored in electronic medical records.

Keywords: PDF, Acquired, Data Accuracy, Data Format

1. 緒論

電子カルテの普及に伴い、多施設の電子カルテデータの臨床研究等への二次利用が取り組まれている。MID-NET（Medical Information Database Network）は国の医療情報データベース基盤整備事業で電子カルテやレセプトデータを匿名化して収集しているが、その対象は処方・注射オーダや検体検査情報など、電子カルテ上で既に構造化され蓄積されている情報に限定されている [1)]。一方、臨床研究での二次利用を考えると、電子カルテに記載する経過記録等のデータを構造化して取得する必要がある。診療録直結型全国糖尿病データベース事業（J-DREAMS）では、処方・注射オーダ等に加え、医師が記載する経過記録を、テンプレートを使用することで構造化して、データ収集している [2)]。我々は、電子カルテに電子症例報告書システムを組み込み、テンプレートを用いて診察記事を記載しながら症例データを蓄積するシステムを構築し、電子カルテメーカー4 社の電子カルテを導入する 15 病院に展開している [3)]。これらの構造化データ取得の取り組みは前向き研究で可能となる。一方、ビッグデータを用いた臨床研究を考えると、電子カルテに蓄積されたデータを用いた後ろ向き研究を考慮する必要がある。このためには、電子カルテに蓄積された非構造化データの利活用が課題となる。

電子カルテには、レセプト情報や様々な検査レポート、病院受診時の診療記録などの診療データが保存されている。これらの膨大なデータを、多施設参加型の観察研究で活用することが期待されているものの、担当ベンダー間のギャップが原因で、一括で診療情報を集積し構造化することは困難である。しかしながら、文書共有を目的として開発され、国際標準化された PDF (Portable Document Format)形式であればデータの収集に関する多施設間の問題は容易に解決することが出来、残る課題は PDF 文書からの情報抽出と構造化のみとなる。

PDF 文書は意図したレイアウトでどのデバイスでも正しく表示されるように規格化されているが、内部データは表示位置に従って配列されるとは限らないため、その情報抽出と構造化は容易ではない。しかしながら、PDF 形式で保存された診療データの中でも、検査レポートは定型的な書式で出力されていることが多いため、他の非構造化データよりも情報抽出が実現出来る見込みは高いと言える。検査レポートは電子カルテ内に膨大に存在する。ここから臨床研究に有用な計測値を取得することが出来れば、臨床研究で利用できるデータ項目を大きく拡張させることができる。

2. 目的

PDF 形式で保存された検査レポートから、臨床研究に必要なデータを取得して構造化するプログラムを作成し、その抽出精度を検証する。

3. 方法

3.1 対象

大阪大学医学部附属病院で作成された放射線検査レポートと呼吸機能検査レポートから、それぞれ 100 枚、125 枚を学術研究目的で抽出した。そのうち各 10 枚を評価用データとして、それら以外を抽出ルール作成のための学習データと

して使用した。

3.2 システムの概要

本システムの開発には Java と JavaScript を使用した。PDF 文書に対する操作に Java ライブラリ「Apache PDFBox（以下、PDFBox と表記）」を用いた。座標取得用の html インターフェースには Vue.js を用いて構築した。

本システムの概要を図1に示す。全 5 ステップから成り、以下にその詳細を説明する。

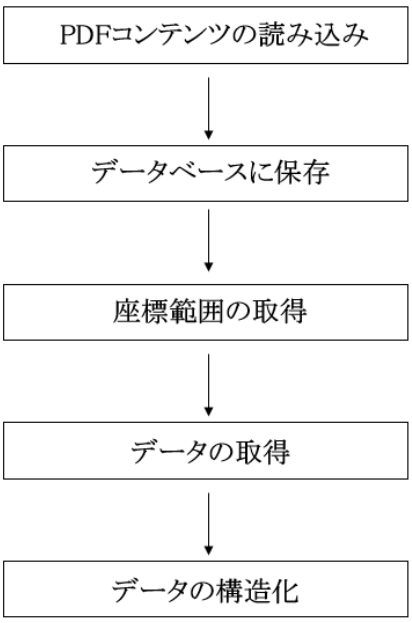

図1 システムの概要

3.2 PDF コンテンツの読み込み

PDF 文書内部は、文字列とその4つの頂点の座標がページの左隅を原点座標（0, 0）として定義される。文字列は先に Y 軸座標値の降順、次に X 軸座標値の昇順で配置されている。

PDF 文書を PDFBox で処理すると、文字列のテキストとそれに対応する4つの頂点の座標を取得することができる。PDF 文書のテキストデータは文字単位で 1 つずつ出力されるため、最小ピクセル間隔を設定して文字列を適切な範囲で結合・分割する前処理が必要となる。本システムでは、最小ピクセル間隔を 12px に設定した。

3.3 データベースに保存

PDF 形式で保存された文書内には本研究の対象であるテキスト以外にも、画像などのデータも格納されている。そこで、PDF 文書に対して一括で 3.1 の処理結果を事前に適用してデータベースに予め保存することで、データの読み込みにかかる負荷を軽減させられるように設計した。保存される項目には、抽出されたテキストとその位置を定義する座標（X 軸左右端、Y 軸上下端）を選択した。PDF 文書におけるテキストの配置をベースに、それらを上から下、左から右の順にデータベースに保存した。このように保存することで、以降のステップでは、データベースのテーブルを読み込んで、対応するデータを取得することができる。

3.4 座標範囲の取得

PDF 文書をブラウザに表示させ、その上でユーザーが矩形領域を指定する html インターフェースを構築した。対象データの座標範囲として、1 つ以上の矩形領域を指定する。ここで指定した座標範囲は次のステップのデータの取得に使用される。同種の検査レポートは全て同じ書式であり、対象データの座標範囲も同一になっている。従って、各種類の PDF 文書に対して初回だけ座標を指定すればよい。

座標範囲を指定する際に、その座標範囲に対して分類を与える。まず一つ目には、key-value 形式と（図 2a）、もう一つには、table 形式と定義した（図 2b）。このような分類を適用することで、異なるフォーマットのデータに対して、異なるルールを使って構造化することが可能になる。ルールの詳細については、次項で説明する。

図 2a 検査報告書の key-value 形式のデータ

	予測	実測	%	単位
	フローボリューム			
FVC	2.21	2.45	111.1	L
FEV1.0	1.80	1.66	92.3	L
FEV1.0%		67.8		%
FEV1.0/VC.p		74.2		%
PEFR	5.93	5.24	88.4	L/s
MMF	2.35	0.88	37.7	L/s
V75		4.40		L/s
V50	2.76	1.22	44.0	L/s
V25	0.99	0.31	31.2	L/s
ATI				
COPD_G	GOLD 1:Mild			
	FVC Ps/Pr			
	Pre	Post	Ps/Pr	単位
FVC	2.45			L
FEV1.0	1.66			L
FEV1.0%	67.8			%
PEFR	5.24			L/s

図 2b 検査報告書の table 形式のデータ

3.5 データの取得

3.5.1 key-value 形式のデータの取得

対象データが key-value 形式である場合、全 PDF 文書中で同じ位置に出現する確率が 90%以上の文字列を key として利用した。図 2a を例に説明すると、「身長」や「体重」などが同じ位置に高頻度で出現するため、key と設定出来る。

次に、同じ行の高さに出現し、key1 と key2 の間に存在する文字列を key1 の値として認識するように設計した。例えば、図 2a において、「身長」が「key1」、「体重」が「key2」の場合、この 2 つのテキストの間にある文字列「146.0cm」は「key1」の値として認識されるようにした。

3.5.2 table 形式のデータの取得

図 2b のような、検査結果が表形式で示されるものを table 形式のデータとして扱い、具体的なデータ取得手法を本項で説明する。一般的に、表の見出し行（以下、Row_key）は最左部に、表の見出し列（以下、Column_key）は最上部に位置する。前ステップで table 形式と分類されたデータ範囲に対し、データを上から下方向へ読み出し、その文字列が不変かつ

最上部に位置し、その上で同じ行の位置に複数そのような文字列が存在するデータを、Column_key と認識させる。図 2b を例にすると、「予測」、「実測」、「%」、「単位」が Column_key として取得できる。次に、データを左から右方向へ読み出し、Column_key の認識と同様の処理で、最も左に位置している不変の文字列群を Row_key として特定した。図 2b において、一番左に位置する「FVC」、「FEV1.0」などの文字列が Row_key として特定出来る。それら以外のデータに関しては、ボディとして認識され、データベースに記録される。

3.6 データの構造化

key-value 形式に関しては、JSON 自体が key-value をサポートしている。そのため、この形式で取得されたデータは直接 JSON 形式に変換される(図 3)。


```
{
 "コード" : "00000000",
 "測定日" : "2017/07/0709:05",
 "前回日" : "",
 "温度" : "25.0℃",
 "湿度" : "40.1%",
 "気圧" : "752mmHg",
 "氏名" : "患者0000",
 "年齢" : "66歳",
 "性別" : "男性",
 "身長" : "170.0cm",
 "体重" : "65.0kg",
 "BSA" : "1.75㎡",
 "医師名" : "医師0000",
 "検者名" : "検者　0000",
 "依頼科" : "消外",
 "病棟" : "外来",
 "検査種別" : "9D100",
 "ｺﾒﾝﾄ1" : "",
 "ｺﾒﾝﾄ2" : "",
}
```


図 3　JSON 形式で保存した key-value 形式のデータ

table 形式のデータに対する構造化処理を、図 2b を例に説明する。Row_key「FVC」の行に存在する値の構造化を考える。まず、対象データに対応する Column_key の文字列「予測」、「実測」、「%」、「単位」、「FVC」をマーカーとして、各々の座標値をデータベースから取得する。次に、「予測」の Y 軸座標値と「FVC」の X 軸座標値に位置する文字列「2.21」は対象データとして取得される。同様に、「実測」の Y 軸座標値と「FVC」の X 軸座標値に位置する文字列「2.45」が取得出来る。このように、Row_key と Column_key の対応関係を基に、JSON 形式で構造化することが可能である(図 4)。


```
▼{
 ▼FVC: {
    予測: "2.92",
    実測: " ",
    %: " ",
    単位: "L"
 },
 ▼FEV1.0: {
    予測: "2.08",
    実測: " ",
    %: " ",
    単位: "L"
 },
 ▼FEV1.0%: {
    予測: "71.2",
    実測: " ",
    %: " ",
    単位: "%"
 },
 ▼FEV1.0/VC.p: {
    予測: " ",
    実測: "60.5",
    %: " ",
    単位: "%"
 },
 ▼PEFR: {
    予測: "7.68",
    実測: " ",
    %: " ",
    単位: "L/s"
 },
```


図 4　JSON 形式で構造化した table 形式のデータ(一部)

3.7 精度評価

JSON 形式として正しく構造化が出来たかどうかを軸に、その精度を算出する。まず、JSON の key に対応する文字列が正しく取得できた割合を示す。次に、key が正しく取得できたものを対象として、value も正しく取得出来た項目の割合を確認する。なお、key が正しく取れていない場合は欠損値となり、失敗事例として扱う。

精度評価に用いるデータとして、放射線検査レポートと呼吸機能検査レポートから、ルール構築に使用しなかった各 10 枚(計 20 枚)を使用した。

4 結果

key-value 形式のデータにおいては、key、value のいずれも図 3 のように正確に取得して構造化することが出来た。

Table 形式のデータでは、key の取得精度は 90.9% (80/88)であった。key は正しく取得できたが value に間違いのある事例はそのうち、81.8% (72/88)であった。実際の構造化事例を図 4 で示す。Row_key、Column_key、value の全てが正しく認識できている場合には、問題なく JSON 形式で構造化することが出来た。Row_key は認識出ているが Column_key の認識に失敗している場合では、対応する value が認識していたとしても構造化に失敗した。例えば、Row_key「FVC」は認識できているが Column_Key「Pre」は認識出来ていないため、データベースに保存されている value「3.02」を取り出すことは出来ない。その結果、FVC-Pre に該当する key とその value は欠落した。

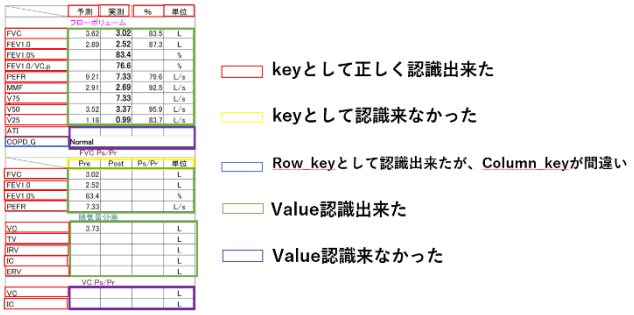

図 4　Table 形式のデータ構造化結果

5 考察

本研究では多施設から検査レポートを収集して解析し、将来の臨床研究で利用できるように構造化することを想定し、PDF で保存された検査レポートからデータの抽出と構造化を実現するプログラムを開発した。


```
▼ATI: {
    予測: " ",
    実測: " ",
    %: " ",
    単位: " "
},
▼COPD_G: {
    予測: " ",
    実測: " ",
    %: " ",
    単位: " "
},
```


図 5　間違った構造化データ

key-value 形式のデータにおいては問題なく構造化を達成することが出来たが、table 形式のデータでは key の抽出、value の取得のいずれにおいても欠損が出現した。取得に失敗した事例をいくつか示す。図 5 は、Row_key「COPD_G」に対応する値の取得失敗事例である(抽出対象の表は図 2b)。本来であれば、COPD の重症度を示す「GOLD 1:Mild」が取得されるべきであったが、「GOLD 1:Mild」の X 軸座標値の幅が複数の Column_key の X 軸座標にまたがるため、正しく取得されずに欠損値となった(正しく取得された場合、Column_key なしで構造化されるべきである)。また、表組みの複雑さにより誤って構造化された箇所も存在した。例えば図 2b における「フローボリューム」と 「FVC Ps/Pr」、そして「FVC Ps/Pr」の直下に存在する「Pre」「Post」「Ps/Pr」「単位」が該当する。「フローボリューム」と 「FVC Ps/Pr」に関しては、いわば表のタイトルというべき見出しに該当し、複数の表で表現できるものを敢えて結合して一つの表として扱った結果、表の中段にタイトルが出現している。「FVC Ps/Pr」の直下の「Pre」「Post」「Ps/Pr」「単位」は、別表であり改めて Column_key に該当するものを定義する形となっている。そのため、「表の最上段を Column_key」とした本研究のルールでは対応できない項目になっていた。

現在のシステムで対応できるのは、比較的簡単な書式のテーブルデータの抽出と構造化までとなっている。本事例のような、複数の表が結合した複雑なテーブルの場合、現在の方法では一部のデータの構造化に失敗してしまう。対応出来る table 形式の幅を広げるためには、別の精度を上げる手法を検討する必要がある。例えば複雑なテーブルに対して、全体の座標範囲だけでなく Column_key と Row_key になるの座標範囲もマークしておけば、より良い構造化が可能となる。

分析対象のレポート構造は医療施設間で異なることが想定される。本手法ではそれぞれの医療施設のレポートを分析する際、最初に html インターフェースで目標データ範囲を設定する必要がある。将来的には、機械学習の画像認識技術を応用して、PDF 文書中のテーブル領域が自動認識出来るように本システムを拡張することを検討している。目標範囲を自動的に取得することが可能になれば、PDF 文書からのデータ抽出とそれによる多施設からの検査値収集がより簡便になり、臨床研究を促進させる効果が期待できる。

6 結論

電子カルテに蓄積された PDF 形式の検査レポートからデータを抽出して構造化することが可能であることを確認した。現在のシステムでは複雑な表組みに対しては限界が存在するが、key の認識ルールを調整して JSON 構造を再定義することで、対応出来る可能性がある。

参考文献

1) 大江 和彦. MID-NET:医薬品安全対策のための医療情報データベース. 生体医工学 2017; 55(4) :159-164.

2) 診療録直結型全国糖尿病データベース事業 http://jdreams.jp/

3) Matsumura Y, Hattori A, Manabe S, Tsuda T, Takeda T, Okada K, Murata T, Mihara N. A Strategy for Reusing the Data of Electronic Medical Record Systems for Clinical Research. Stud Health Technol Inform. 2016; 228: 297-301..