RD-002

多重仮説文書構造ネットワークを用いたデータ抽出方式の開発

Development of Data Extraction Method using Document Network Structure with Multiple Hypotheses

関 峰伸† 小林 義行† 芳賀憲行‡ 石田響子‡ 藤尾正和† 平山淳一† 永崎健†

Minenobu Seki Yoshiyuki Kobayashi Noriyuki Haga Kyoko Ishida Masakazu Fujio Junichi Hirayama Takeshi Nagasaki

1. はじめに

高品質な設計を効率的に行うため，過去の仕様書に記載されている仕様データを参照し，再利用することが有効である。本稿では，OCR(Optical Character Recognition)技術と文書構造解析技術を用いて，仕様書から仕様データを自動的に抽出する方式について述べる。本方式では，複数のレイアウト構造を表現する多重仮説文書構造ネットワークを用いてデータを抽出する。また，仕様項目の階層構造とデータの種類が記載されている階層付項目辞書の構造を照合することで，複数の仮説の中から適切なレイアウト構造の解釈を選んで，データを抽出できる。また，各仕様項目に対応する可能性のある複数のデータを候補として抽出し，ユーザにそれらを表示するインターフェイスを持つツールを提案する。第 1 位のデータ候補に誤りがあった場合にも，ユーザは簡単にその他のデータ候補の中から正解データを探すことができる。また，PDF 文書の中の文字と罫線の情報を用いてレイアウト解析を行う。そのため，紙の場合は OCR で認識した結果を PDF 文書へ変換し，電子文書の場合は直接 PDF へ変換し，データを抽出できる。そのため，紙文書も電子文書も区別なく処理することができる。

2. 関連研究

OCR 技術は，文書 OCR と帳票 OCR に分けられる。文書 OCR は，OCR の専門知識がない人でも汎用的に用いることができる OCR である。ただし，文字を抽出し文字コードへ変換するのみである。帳票 OCR は，文字コードへの変換のみでなく，指定したデータを抽出する機能を含む。罫線の情報から枠構造を認識し，文字の位置関係から文字列を抽出し，対象となる文字列をデータとして出力する。しかし，帳票の種類の限定や事前に抽出すべき帳票の項目の定義に専門家の知識が必要である。そのため，帳票 OCR の技術開発は，適用対象拡大と事前定義の簡易化が行われている。適用対象拡大では，大量の書式限定帳票の読み取りから，少量多品種の帳票の読み取りへと適用対象の拡大を行ってきている。具体的には，OCR 専用帳票，一般定型帳票，一般非定型帳票へと読み取り対象を拡大させてきている。事前定義の簡易化では，読み取りのための事前定義の簡易化を行っている。現在の製品では，読み取り位置を絶対座標で定義しておく方式と，枠構造を定義しておく方式が一般的である。しかし，これらの方法では，仕様書のような非定型(非構造化)文書からのデータを抽出することができない。データが記載される位置が不確定であるため，同じ位置，同じ枠にデータが記載されないからである。これに対し，事前に項目名の辞書を定義する方式[4]がある。辞書には，帳票に記載される項目名をすべて列挙する。項目名の位置を特定し，項目名の位置からデータの位置を推定することができる。しかし，文書に記載されるすべての項目名を定義し，項目名の文字列なのかデータの文字列なのかを判定する必要がある。仕様書のように多くの仕様項目が存在し，すべての項目名を定義できない場合，文書構造の解釈に曖昧性が発生し，データ位置の推定精度が低下してしまう。また，項目名の論理的な階層関係と項目名間の相対位置関係を定義する方式[5]がある。この方式では，項目名間の関係情報を利用できるため，文書構造の解釈の曖昧性は減少する。しかし，非定型文書において，事前に項目間の相対位置関係をすべて定義することは難しい。このように，紙文書からデータを抽出する取組みが行われている一方で，近年，PDF 文書の構造解析技術の開発が取り組まれている[4][8][9]。

† 日立製作所中央研究所 Central Research Laboratory, Hitachi, Ltd.

‡ 日立製作所横浜研究所 Yokohama Research Laboratory, Hitachi, Ltd.

3. 仕様データ抽出

3.1 仕様データ抽出の概要

仕様データ抽出技術の概要を述べる。図 1 は仕様データ抽出のイメージ図である。仕様書には各機器に対する型番や電源周波数や重量等の項目名に対応するデータが記載されている。仕様データ抽出とは，文字認識と文書構造解析技術を用いて，機器 A の型番は AAA，機器 A の電源周波数は 50Hz，機器 A の重量は 0.8kg という，項目名とデータの対応関係を抽出することである。

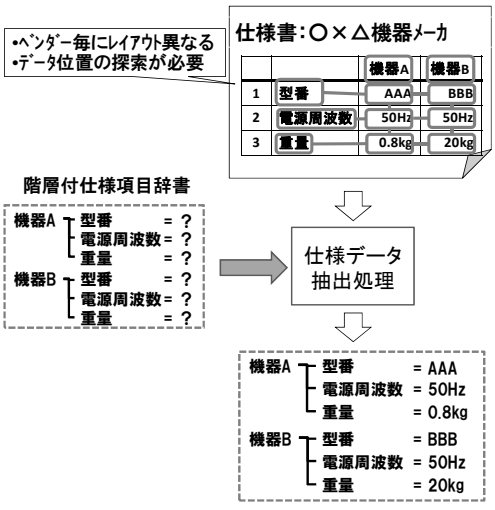

図 1 仕様データ抽出機能のイメージ図

3.2 仕様書における表構造の特徴

仕様書は，様々な会社で独自に作成した文書であり，多種類のデータが記載される。次の 3 つの特徴がある。1 つ目は，仕様データを指し示す項目が階層構造のある複数の項目で記載されていることである（階層有）。図 2(a)に示すように，複数の異なる位置に記載される項目名によってデータの位置が指定される。また，図 2(b)に示すように水平方向からの項目と垂直方向からの項目によって，2 次元構造でデータが指定される。2 つ目は，項目とデータの間に単位を示す文字列が挿まれていることである（単位欄）。図 2(c)に示すように項目とデータの間に単位を示す文字列が挿まれている。3 つ目は，枠線がない，或いは枠を構成する罫線の一部が欠如しているということである（罫線無）。図 2(d)に例を示す。

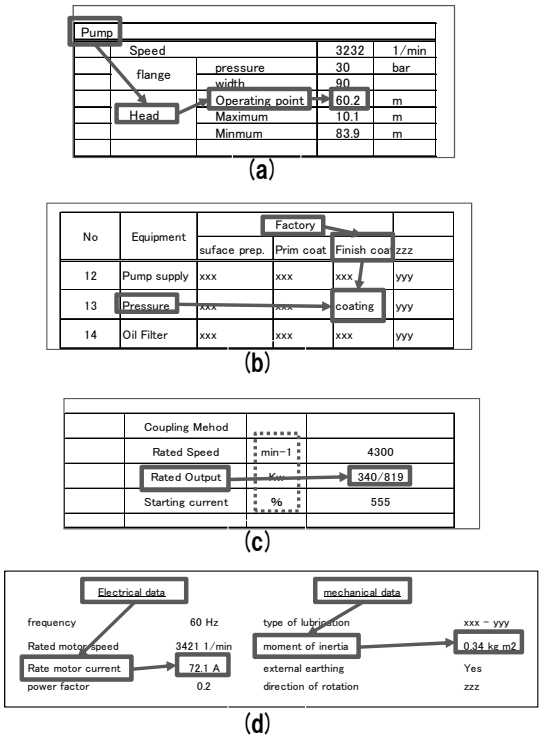

図 2　仕様書の例

表 1　仕様書における表構造の特徴

項目	特徴
① 階層有	仕様データを指し示す項目が階層構造のある複数の項目で記載される。水平方向からの項目と垂直方向からの項目によって，2 次元構造でデータの位置が指定される。
② 単位欄	項目とデータの間に単位を示す文字列が挿まれている。
③ 罫線無	枠がない，枠を構成する罫線の一部が欠如している。

3.3 課題とアプローチ

仕様書には，多種類のデータが存在するため，すべての仕様項目を定義することができず，指定した項目名以外の文字列が項目名なのかデータなのかがわからない。そのため，文書構造の解釈に曖昧性が発生する。また，前述のように階層構造，単位欄，罫線無があると，曖昧性が増加する。例えば，図 3 において，(A)と(B)と(C)の枠の配置は同じである。これに対し，(A)はすべての文字列が垂直方向下向き関係していると解釈した図である。(B)は，1 列目と 2 列目が水平方向左から右へ関係し，3 列目が垂直方向下向きに関係していると解釈した図である。(C)は，1 列目から 3 列名が水平方向左から右へ関係していると解釈した図である。以上のように，同じ枠の配置であっても文書構造の解釈には 3 通り以上の解釈がある。また，(D)と(E)と(F)は同じ枠の配置である。また指定した項目名の位置も同じである。これに対し，(D)は水平方向左から右へ関係していると解釈した図である。(E)は，垂直方向上から下へ関係していると解釈した図である。(F)は，項目とデータの関係に，水平方向と垂直方向の 2 次元の関係があると解釈した図である。このように 2 次元構造への解釈の曖昧性もある。また，様々な会社が自社で作成した文書であるため，事前に項目とデータの物理的な位置関係を定義しておくことができない。

そのため，仕様書データ抽出では，文書構造の解釈に曖昧性が発生する中から，物理的な位置関係を辞書に定義することなく，参照したいデータを抽出することが課題となる。

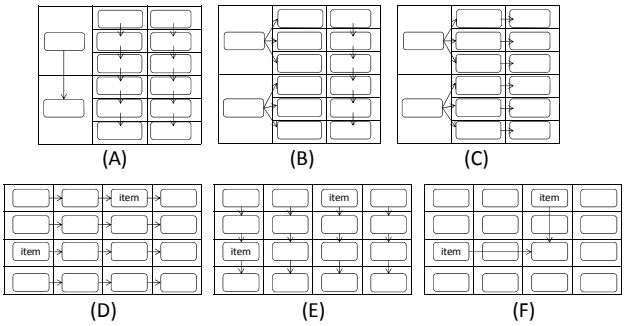

図 3　文書構造の曖昧性

これに対する 1 つ目のアプローチとして，多重仮説文書構造ネットワークと階層項目辞書を用いてデータを抽出する。複数のレイアウト構造を表現する多重仮説文書構造ネットワークを生成し，仕様項目の階層構造とデータの種類が記載されている階層付項目辞書の構造を照合する。これにより，レイアウト構造の曖昧性を低減しながら，データを抽出できる。論理的な階層関係のみを定義する方式である。そのため，専門知識を持たないユーザでも参照したいデータの項目名のみを指定するだけで，辞書を定義することができる。

ただし，上記アプローチを用いたとしても文書構造の曖昧性は残る。そのため，2 つ目のアプローチとして，各仕様項目に対応する可能性のある複数のデータを候補として抽出する。ＧＵＩを介してユーザが候補の中からデータを

選択できるツールを提供する。第 1 位のデータ候補に誤りがあった場合にも，ユーザは簡単にその他のデータ候補の中から正解データを探すことができる。

また，設計業務では紙と電子文書の仕様書が混在する。3 つ目のアプローチとして， PDF 文書の中の文字と罫線の情報を用いてレイアウト解析を行う。そのため，紙の場合は OCR で認識した結果を PDF 文書へ変換し，電子文書の場合は直接 PDF へ変換し，データを抽出できる。

4. 多重仮説文書構造ネットワークを用いたデータ抽出方式

4.1 処理フロー

図 4 に処理フローを示す。仕様データ抽出処理には 7 つのステップがある。

【ステップ 1】文書情報取得処理

文書情報取得処理では，入力文書に含まれる文字の情報（文字種と位置），罫線の情報（位置）を取得する。

【ステップ 2】レイアウト解析処理

レイアウト解析処理では，文字の位置情報と罫線の位置情報を用いて枠の抽出と文字行の抽出を行う。

【ステップ 3】文字列判別処理

文字列判別処理では，文字行内の文字列が何を示す文字列なのか，属性を判別する。具体的には，①階層付項目辞書内の項目名であるのか，②データの種別はなんであるか，③単位文字列であるか，④単位指示文字列であるか，の 4 つ判別を行う。単位文字列とは，“Hz”や“KW”などの単位を表わす文字列である。単位指示文字列とは，“UNIT”や“単位”などの単位を記載する欄を指し示す文字列である。

【ステップ 4】多重仮説文書構造ネットワーク生成処理

多重仮説文書構造ネットワーク生成処理では，複数の文書構造の可能性を表現する多重仮説文書構造ネットワークを生成する。文字列をノードとして論理関係のあるノード間にエッジを形成する有効グラフである。

【ステップ 5】項目データ対応候補生成処理

項目データ対応候補生成処理では，多重仮説文書構造ネットワークから，階層項目辞書の各エントリに該当する項目名とデータの文字列の組(項目データ対応)を抽出する。各エントリに該当する項目名とデータ文字列の対応関係には，複数の対応関係の可能性がある。そのため，項目とデータの対応の候補(項目データ対応候補)を複数抽出する。これを項目データ対応候補と呼ぶ。

【ステップ 6】項目データ対応候補ランキング処理

項目データ対応候補ランキング処理では，コンテンツの知識を用いて，抽出された項目データ対応候補をランキングする。コンテンツの知識として，階層付項目辞書と単位文字列辞書と単位指示文字列辞書を用いる。メインの知識となるのは，階層付項目辞書である。各エントリに対し，各項目データ対応候補がどの程度一致するかの度合い(項目データ対応スコア)を算出し，項目データ対応スコアを用いてランキングする。また，単位に関する知識として，単位文字列辞書と単位指示文字列辞書を用いる。単位文字列と照合された文字列，単位指示文字列と照合された文字列の情報を用いてランキングする。

この処理によって，単位文字列が項目とデータの間に挿まれている場合にも，単位文字列ではなくデータを上位に出力する。

以降の節では，ステップ 3～6 について詳しく述べる。

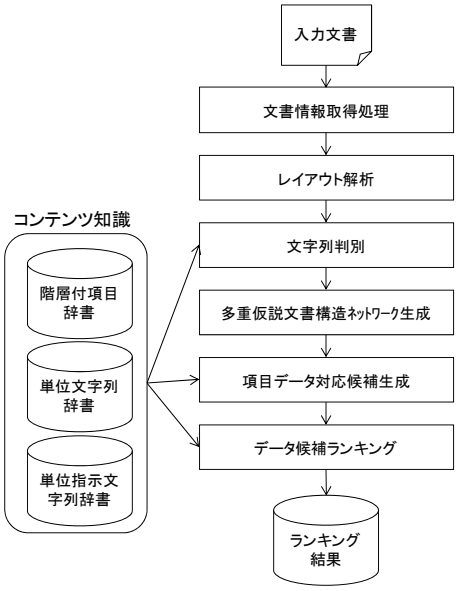

図 4　仕様データ抽出の処理フロー

4.2 文字列判別処理

文字列判別処理では，項目名文字列照合，データ文字列種判別，単位文字列照合，単位指示文字列照合の 4 種類の処理を行う。文字列が項目名，単位文字列，単位指示文字列と一致するか否かの判定には，レーベンシュタイン距離[10]をベースに文字列長を考慮した評価関数を用いた。

表 2　文字列判別処理の内容

処理	内容
項目名文字列照合	文字列が階層付項目辞書内にある項目名と一致するか判定する。一致する文字列を「指定項目文字列」，一致しない文字列を「未決定文字列」とする。未決定文字列には，階層付項目辞書に入っていない項目名を表わす文字列とデータを表わす文字列があり，それらの区別はつかない。
データ文字列種判別	文字列が数字だけで構成される数字列であるのか，文字列が数字以外の文字で構成される非数字文字列であるのか，文字と数字で構成される数字文字列であるのかを判別する。
単位文字列照合	文字列が単位文字列辞書に記載された文字列と一致するかを判定する。
単位指示文字列照合	文字列が単位指示文字列辞書に記載された文字列と一致するかを判定する。

4.3 多重仮説文書構造ネットワーク生成処理

次の特徴を利用して，多重仮説文書構造ネットワークを生成する。1 つ目は，文書に記載される文字列間の論理関係は，左から右，上から下へ意味の結合があるように記載されるという特徴である。2 つ目は，水平方向と垂直方向に整列された文字列間には論理関係があるという特徴である。枠線が存在する場合は，枠端の位置が揃っている枠内の文字列に論理関係がある。

4.3.1 枠端位置を利用した生成方法

図 5 に示すように，入力文書をレイアウト解析すると枠と文字行が抽出される。この枠と文字行の配置関係を解析し，多重仮説文書構造ネットワークを生成する。図 6 に示す(a)と(b)のように１：N（Nは１より大きい整数）の関係で枠端位置が揃う場合，枠内の文字行には項目名とデータ，或いは項目名と項目名の意味的階層関係がある場合が多い。また，図 6 に示す(c)と(d)のように１：１の関係で，枠端位置が揃う場合，枠内の文字列には，項目名とデータ，或いは連続するデータの関係がある場合が多い。そして，文書に記載の文字列は，左から右，上から下へ項目とデータ，項目の上下の関係をもつように記載される。そのため，左から右，上から下へとつながるリンクを生成する。 (a)と(b)の場合と同様に，文書に記載の文字列は，左から右，上から下へ項目とデータ，データの順番の関係を持つように記載されるため，左から右，上から下へのリンクを生成する。また，項目の位置から下方向，或いは右方向へ連続するデータの記載に対応するため，図 7 に示すように，枠端の位置が同じ枠が連続する場合は，連続する複数の枠内の文字列とのリンクを生成する。ハッチングがかかった 2 つの文字列からのリンクについてのみ図示している。他の文字列からも同様に上から下，左から右へリンクが生成される。同様に他の枠内の文字列からもリンクが生成される。これにより，連続するデータと項目の対応付けができる。

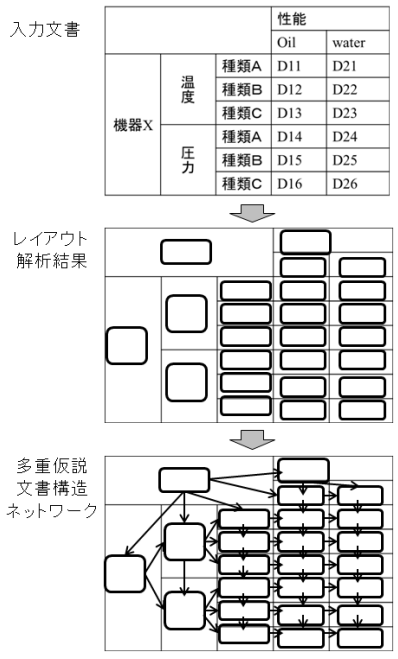

図 5　多重仮説文書構造ネットワークの生成

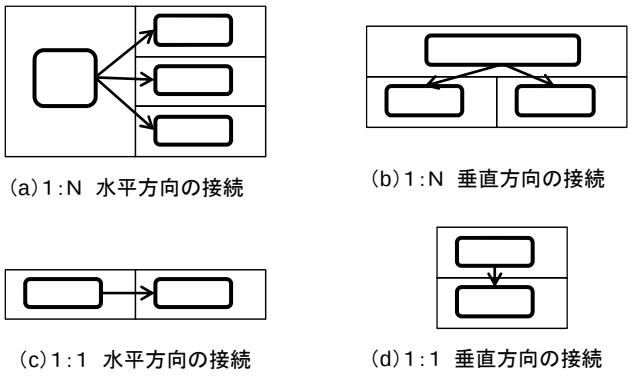

図 6　枠端整列性に基づくネットワーク生成

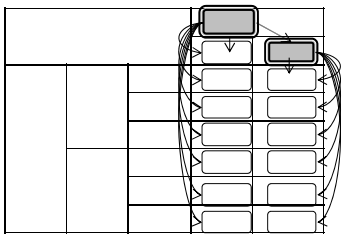

図 7　連続するデータの記載に対応するネットワーク生成

4.3.2 文字列位置を利用した生成方法

これまで入力文書に枠が存在する場合について述べてきた。本節では，枠がない文書，或いは枠を構成する罫線の一部が欠如している文書についても適用可能であることを述べる。枠がない場合，枠端位置の整列性解析を行う代わりに，文字列位置の整列性解析結果を用いることによって多重仮説文書構造ネットワークを生成する。枠がない場合のレイアウト解析処理には，recursive XY cut[11]等のトップダウンの解析方法，文字矩形間の距離を判定して文字矩形を統合していくボトムアップの解析方法，トップダウンの解析方法とボトムアップの解析方法を組み合わせる方法等がある。解析方法やパラメタの違いにより解析結果は異なる。

図 8 に，入力文書に対する 3 種類のレイアウト解析結果を示す。レイアウト解析結果 A は，行方向(水平方向)を優先して矩形を統合したレイアウト解析結果である。レイアウト解析結果Ｂは，行方向だけでなく列方向(垂直方向)の分割を行ったレイアウト解析結果である。レイアウト解析結果Ｃは，レイアウト解析結果Ｂの方式に比べて垂直方向の分割が優位に働くパラメタで解析した結果となっている。各レイアウト解析結果の中にあるブロック内の文字列同士にはリンク関係がある。図 8 の文書構造ネットワークＡ～Ｃは，レイアウト解析結果Ａ～Ｃの論理構造を示している。具体的には，文書構造ネットワークＡでは，同じブロック内にある文字列 BBB から文字列 EEE をリンクする。同様に，文字列 CCC から文字列 DDD，文字列 DDD から文字列 FFF，文字列 FFF から文字列 GGG，文字列 xxx から文字列 yyy，文字列 yyy から文字列 zzz，文字列 zzz から文字列 qqq をリンクする。また，ブロック間のリンクのため，先頭文字列間を上から下に向かってリンクする。

図 9 は，上記のように生成された複数の文書構造ネットワークから階層付項目辞書を用いて項目データ対応候補を生成した結果である。文書構造ネットワークＡでは，文字列 AAA から文字列 BBB の関係までしか辿ることができない。文書構造ネットワークＣでは，文字列 AAA から文字列 BBB，文字列 BBB から文字列 CCC，文字列 CCC から文字列 XXX と辿ることができる。その結果，文字列 AAA と文字列 BBB と文字列 CCC を項目名，文字列 xxx をデータとする項目データ対応候補を生成する。

図 10 は，図 8 で示した複数の文書構造ネットワークの論理和をとったネットワーク（多重仮説文書構造ネットワーク）である。この多重仮説文書構造ネットワークにおいて，正しい項目データ対応候補を生成することができる。

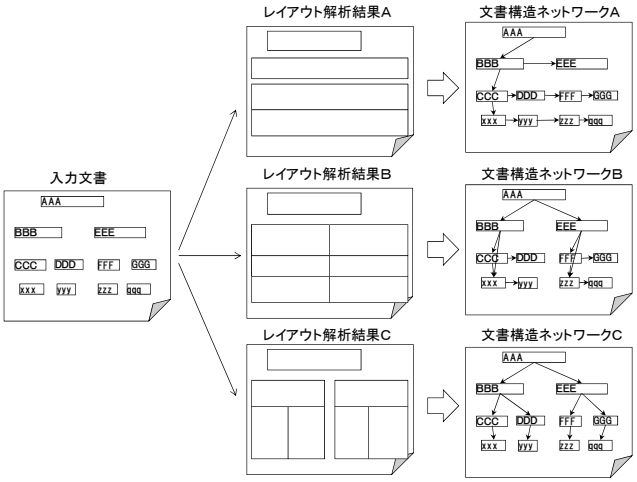

図 8　枠がない場合のレイアウト解析処理結果

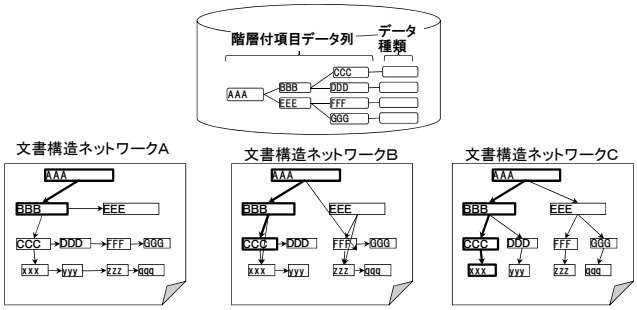

図 9　枠がない場合の項目データ対応候補生成結果

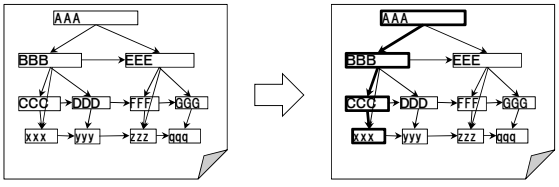

図 10　枠がない場合の多重仮説文書構造ネットワーク

4.4 項目データ対応候補生成処理

項目データ対応候補生成処理では，多重仮説文書構造ネットワークから複数の項目データ対応候補を生成する。図 11 に示すように，階層付項目辞書のすべてのエントリに対し，全ての非指定項目文字列を起点とする探索処理を行う。図 12 に示すように，探索処理では，起点としている非指定項目文字列がデータであると仮定し，非指定項目文字列とリンクする項目名文字列を探索する。まず左方向に出現する項目名文字列を探索する。次に上方向に出現する項目名文字列を探索する。その結果得られた左方向探索結果と上方向探索結果を連結することで項目データ対応候補とする。

図 13 の(a)にハッチングで示す文字列は，itemZ と itemA と itemB が項目名として照合された場合に候補となる非指定項目文字列である。図 14 に正解となる項目データ対応候補を示す。階層付項目辞書の中で着目中のエントリにある 3 つの項目名が一致する非指定項目文字列である。

図 13 の(b)は，(a)と文字列の配置が異なる表である。ハッチングで示す文字列は itemA と itemB が項目名として照合された場合に候補となる非指定項目文字列である。図 15 に正解となる項目データ対応候補を示す。左方向の探索結果と上方向の探索結果を連結することにより，2 次元の項目名で指定された非指定項目文字列を抽出する。

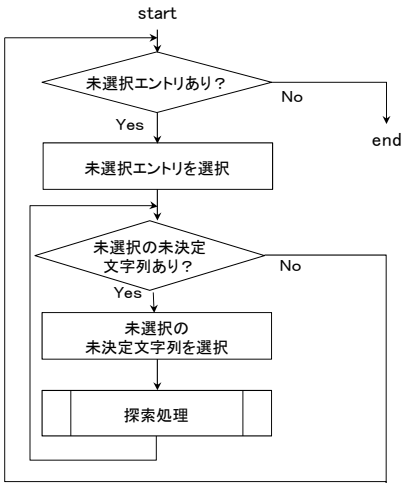

図 11　項目データ対応候補生成処理

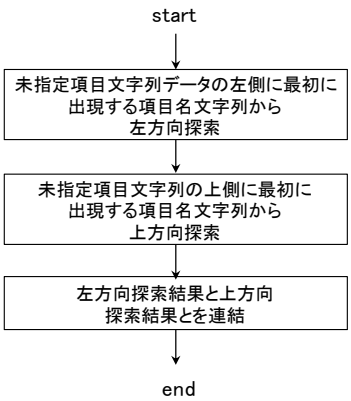

図 12　探索処理

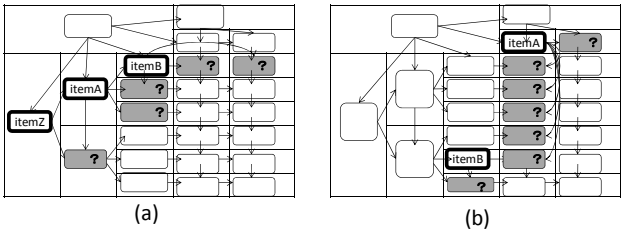

図 13　複数の項目データ対応候補の生成

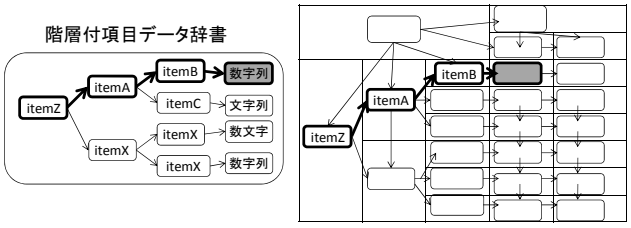

図 14　正解の項目データ対応候補(a)

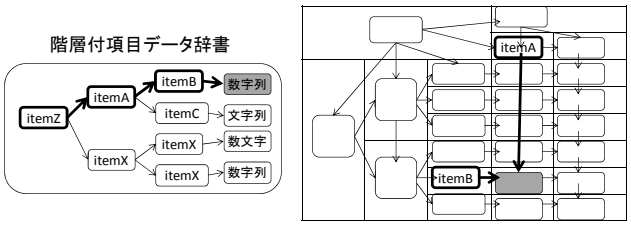

図 15　正解の項目データ対応候補(b)

4.5 項目データ対応候補ランキング処理

項目データ対応候補ランキング処理では，コンテンツの知識を用いて，抽出された項目データ対応候補をランキングする。コンテンツの知識として，階層付項目辞書と単位文字列辞書と単位指示文字列辞書を用いる。メインの知識となるのは，階層付項目辞書である。各エントリに対し，各項目データ対応候補がどの程度一致するかの度合い(項目データ対応スコア)を算出し，項目データ対応スコアを用いてランキングする。また，単位文字列辞書と単位指示文字列辞書を用いて，正しいデータを上位にランキングさせる。

4.5.1 階層項目辞書の階層関係を用いたランキング

データ候補ランキング処理では，階層付項目辞書の各エントリに対し，項目データ対応候補がどの程度一致するか（項目データ対応スコア）を算出し，項目データ対応候補をランキングする。図 16 は，各エントリに対して複数の項目データ対応候補がランキングされた結果のイメージ図である。項目データ対応スコアは，次の 5 つの値の重み付き線形和となる。

(1)　項目名の一致数：項目データ対応候補の中にある項目名の中で，着目しているエントリ内の項目名と一致する数

(2)　項目名不一致数：項目データ対応候補の中にある項目名の中で，着目エントリ内の項目名と一致せず，他のエントリ内の項目名と一致する数。マイナスとして用いる。

(3)　項目名照合度：項目名と一致した度合い，レーベンシュタイン距離をベースに文字列長を考慮した値

(4)　項目名順序：着目しているエントリ内の項目名の出現順序と項目データ対応候補内の項目名の出現順序の一致度

(5)　データ一致度：着目しているエントリにおけるデータの種類と項目データ対応候補におけるデータの種類が一致するか

また，項目データ対応候補の中で，データに直接接続されている項目名が，各エントリの中で最下層の項目名と一致する候補を優先して上位にランキングする。これは，各エントリ内に記載される項目名のうち上位の項目名は下位の項目名を修飾している単語となり，最下層に記載される項目名がデータを直接指し示す単語である場合が多いためである。

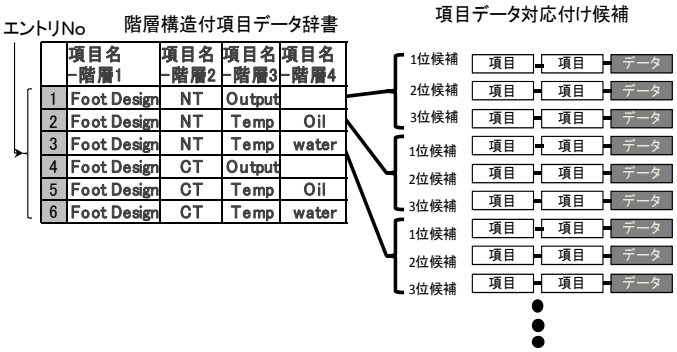

図 16　データ候補ランキング処理結果のイメージ図

4.5.2 単位文字列・単位指示文字列辞書を用いたランキング

単位文字列・単位指示文字列辞書を用いて，項目とデータの間に単位文字列が存在する場合であっても，正しいデータを上位にランキングさせる。

単位文字列辞書を用いたランキング処理では，階層付項目データ辞書の各エントリに対応する複数の項目データ対応候補の中で，単位文字列がデータとなっている項目データ対応候補の順位を下げる処理を行う。図 17 に示すケースでは，単位を示す文字列“KW”と“350”の両方が候補として抽出される。これに対し，“KW”をデータとして持つ項目データ対応候補の順位を下げることにより，“350”をデータとして持つ項目データ対応候補が上位にランキングされる。

単位指示文字列辞書を用いたランキング処理では，階層付項目データ辞書の各エントリに対応する複数の項目データ対応候補の中で，単位指示文字列に記載された文字列が項目名として抽出されている項目データ対応候補の順位を下げる処理を行う。図 18 に示すケースでは，単位を示す文字列“KW”と“350”の両方が候補として抽出される。これに対し，“UNIT”を項目名として持つ項目データ対応候補の順位を下げることにより，“350”をデータとして持つ項目データ対応候補が上位にランキングされる。

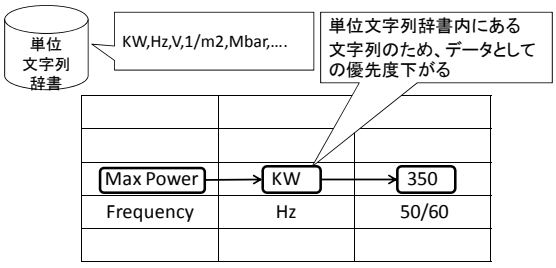

図 17　単位文字列辞書を用いたランキング処理

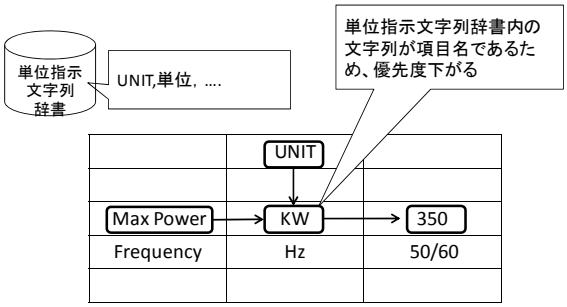

図 18　単位指示文字列辞書を用いたランキング処理

5. 仕様データ抽出ツールの開発

以上で述べた仕様データ抽出方式を用いて，仕様データ抽出ツールを開発した。階層付項目辞書の各エントリに対し，第 1 位のデータ候補に誤りがあった場合にも，ユーザは下位の候補の中から正解となるデータ候補を探すことができる。前処理を事前に処理しておくことによって，ユーザは待ち時間なく，データの確認・修正・登録作業のみを連続で行うことができる。スキャン文書を OCR した結果だけでなく，電子文書を直接 PDF 化したファイルからもデータを抽出できる。

図 19 は，仕様データ抽出作業フローの概略を示す図である。ユーザは，仕様書と仕様書に対応する辞書を指定し，OCR 等の前処理と仕様データ抽出処理を実行させる。次に，仕様データ抽出ツールを用いて，インタラクティブに仕様データ抽出結果を確認し，修正する作業を行う。最後に，すべての仕様項目に対するデータを確認後，ボタン操作等により仕様値 DB へデータを送付する。図 20 は，仕様データ抽出ツールのインターフェイスである。抽出項目表示・データ選択・データ手入力部と仕様書画像表示部に分かれている。プルダウンにある複数の候補の中から正しい候補を選択することができる。また，選択した項目名とデータの対応関係を表示する。階層付項目辞書編集機能と同義語辞書編集機能を持っている。

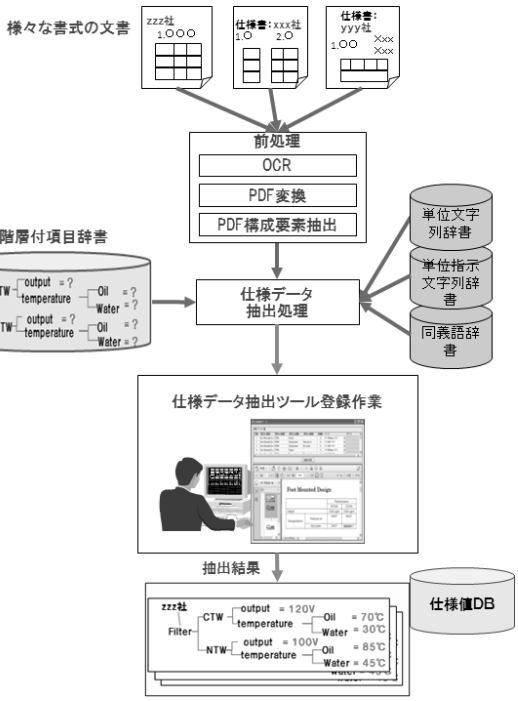

図 19　仕様データ抽出作業フロー

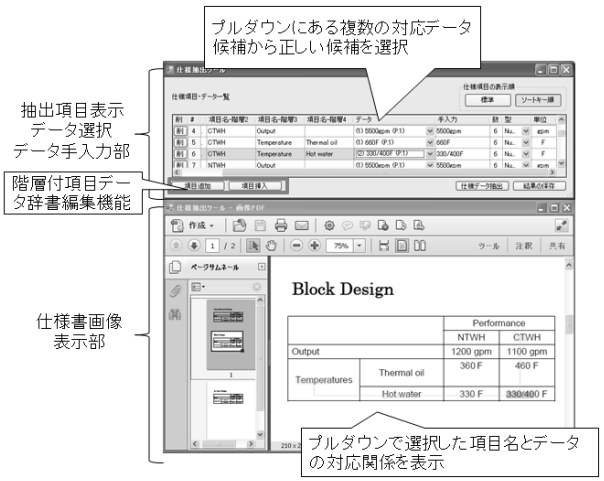

図 20　仕様データ抽出ツールインターフェイス

6. 評価及び効果

6.1 評価対象

評価対象として用いた仕様書サンプルの数は 43 件である。設計に重要な仕様項目の辞書を作成し，上記仕様書からの抽出実験を行った。手書きの文字やマークを多く含む文書，傾きが大きい，ノイズが多いサンプルを除いた。評価には，階層付項目データ辞書の各エントリに対して，正しいデータが，1 位～5 位の候補に入っている割合を用いた。これを累積正解率と呼ぶことにする。(1)階層有(階層構造のある項目で指定される仕様データ)，(2)単位欄(項目と仕様データの間に単位欄が存在する欄に記載された仕様データ) (3)罫線無(罫線無のある欄に記載された仕様データ)，の 3 種類に属する仕様データの精度を算出した。

6.2 評価結果

上記仕 様書に記載されている全仕様データを調査したところ，上記 3 種類の仕様データ件数は，(1)階層有：1122，(3)単位欄：490，(2)罫線無：849 であった。また，その中で重要仕様である仕様データ数は，(1)階層有：166，(3)単位欄：82 ，(3)罫線無：116 であった。仕様データ抽出ツールを用いて重要仕様の抽出実験を行った。その結果，累積正解率は，階層有 85.5%，単位欄有:97.6%，罫線無 85.3%となった。誤抽出の主な原因は，項目名の区切り位置の抽出失敗と文字認識誤りによる項目名照合の失敗であった。

表 3　累積正解率

	仕様項目数	重要仕様項目数	正解数	正解率
階層有	1122	166	142	85. 5%
罫線無	849	116	99	85. 3%
単位欄	490	82	80	97. 6%

6.3 効果及び考察

データ抽出ツールを用いてデータ入力した場合とすべて手作業で入力した場合のデータ入力作業時間を比較した。被験者数は 3 人である。上記 43 件の仕様書の中から 16 件の仕様書を用いた。累積正解率は 84.7%である。1 人あたり，12 項目の仕様データを入力する作業に対して，ツールを用いた場合と全て手入力による場合を 2 回ずつ実施した。手作業の場合とツールを用いた場合を公平に評価するため，それぞれ交互に異なるサンプルで実施した。また，仕様書内の 36 項目を入力する作業を同様に行った。すなわち一人あたりツールによる入力作業と手入力による入力作業を 4 回ずつ行った。この際，入力されるデータの順番と仕様書に記載されるデータとは一致しないように，入力されるデータの順番を入れ替えた。被験者によるばらつきがあるが，1/3～2/3 の入力時間短縮を確認できた。

表 4　手入力によるデータ入力時間

	被験者 1	被験者 2	被験者 3
12 項目	05 分 17 秒	04 分 19 秒	04 分 59 秒
	03 分 30 秒	03 分 45 秒	03 分 21 秒
36 項目	16 分 10 秒	10 分 19 秒	11 分 45 秒
	12 分 36 秒	13 分 27 秒	11 分 57 秒

表 5　データ抽出ツールを用いた場合のデータ入力時間

	被験者 1	被験者 2	被験者 3
12 項目	01 分 25 秒	01 分 45 秒	03 分 12 秒
	01 分 05 秒	01 分 31 秒	02 分 03 秒
36 項目	05 分 54 秒	06 分 45 秒	04 分 42 秒
	05 分 02 秒	05 分 41 秒	08 分 16 秒

7. 結　言

仕様書からのデータ抽出方式を開発した。本方式では，複数のレイアウト構造を表現する多重仮説文書構造ネットワークと，仕様項目の階層構造とデータの種類が記載されている階層付項目辞書の構造を照合することでデータを抽出する。これにより，レイアウト構造の曖昧性を低減しながら，階層構造をもつ仕様項目とデータの位置及び関係を抽出できる。また，各仕様項目に対応する可能性のある複数のデータを候補として抽出し，ユーザにそれらを表示するインターフェイスを持つツールを提案する。第 1 位のデータ候補に誤りがあった場合にも，ユーザは簡単にその他のデータ候補の中から正解データを探すことができる。また，PDF 文書の中の文字と罫線の情報を用いてレイアウト解析を行う。そのため，紙の場合は OCR で認識した結果を PDF 文書へ変換し，電子文書の場合は直接 PDF へ変換し，データを抽出できる。仕様書 43 件を用いて実験を行った結果，累積正解率は，階層有 85.5%，単位欄 97.6%，罫線無 85.3%が得られた。データ抽出ツールを用いてデータ入力した場合とすべて手作業で入力した場合のデータ入力作業時間を比較した結果，累積正解率が 84.7%の場合，入力作業時間が 1/3～2/3 となった。今後の課題として，項目名の区切り位置検出の精度向上を行う。

参考文献

[1]Hiroshi Sako， Minenobu Seki， Naohiro Furukawa， Hisashi Ikeda, Atsuhiro Imaizumi ，” Form Reading based on Form-type Identification and Form-data Recognition，”ICDAR 2003:pp.926-931

[2]Hiroshi Shinjo， Eiichi Hadano， Katsumi Marukawa， Yoshihiro Shima， Hiroshi Sako，”A Recursive Analysis for Form Cell Recognition，”ICDAR 2001:pp.694-698

[3]Yasuto Ishitani，”Document Transformation System from Papers to XML Data Based on Pivot XML Document Method，”ICDAR 2003: pp250-255

[4]Minenobu Seki， Masakazu Fujio， Takeshi Nagasaki， Hiroshi Shinjo， Katsumi Marukawa，” Information Management System Using Structure Analysis of Paper/Electronic Documents and Its Applications，”ICDAR 2007: pp.689-693

[5]A. Minagawa， Y. Fujii， H. Takebe， and K.Fujimoto，”Logical Structure Analysis for Form Images with Arbitrary Layout by Belief Propagation，”ICDAR2007: pp.714-718

[6]Yasuto Ishitani， Kosei Fume， Kazuo Sumita，”Table Structure Analysis Based on Cell Classification and Cell Modification for XML Document Transformation，”ICDAR 2005:pp 1247-1252

[7]Junichi Hirayama， Hiroshi Shinjo， Toshikazu Takahashi， Takeshi Nagasaki ，” Development of Template-Free Form Recognition System，”ICDAR 2011:pp.237-241

[8]関峰伸，永崎健，丸川勝美，”文書構造要約化による情報提供システム，”FIT2006

[9]Jing Fang, Liangcai Gaoa，Kun Bai，Ruiheng Qiu，Xin Tao, Zhi Tang，”A Table Detection Method for Multipage PDF Documents via Visual Seperators and Tabular Structures，”ICDAR2011:pp.779-783

[10] 馬場謙介，于雲青，村上和彰，”ビットパラレル手法によるアライメントアルゴリズム，” 情報処理学会論文誌， 2005, pp80-87

[11] G.Nagy and S.Seth,”Hierarchical representation of optically scanned documents”, in Proc. 7 ICPR, 1984,pp 347-349.