ビューに基づくデータクリーニング方式の提案

大森　弘樹† 　清水　敏之†† 　吉川　正俊††

† 京都大学大学院情報学研究科　〒 606–8501 京都府京都市左京区吉田本町
†† 京都大学大学院情報学研究科　〒 606–8501 京都府京都市左京区吉田本町
E-mail: †hiroki@db.soc.i.kyoto-u.ac.jp, ††{tshimizu,yoshikawa}@i.kyoto-u.ac.jp

あらまし　科学データの分析や検索などを行う際，そのデータがどういうものかを説明するメタデータを利用することが重要になる場合がある．データ提供者・管理者が人手で作成するメタデータは，自由入力による表記ゆれや誤記，記入漏れの多い汚れた情報になりがちである．この際，専門性が高い科学データでは，適切な辞書が存在しなかったり，分野独自の略称が用いられていたりするため，誤ったデータを機械的に修正することは困難である．本研究は，関係データベースに格納された科学データのメタデータなど，機械的な修正が困難なデータを対象にし，問題があると思われる部分を含んだビューをデータ管理者に提示していくことで，管理者に修正方針の判断を委ねつつ効率よくデータクリーニングを行う方式を提案する．さらに，データクリーニングの過程において収集したデータ修正についての知識を用いて，その後のデータ管理に役立てていく手法を議論する．

キーワード　関係データベース, データクリーニング, ビュー, 関数従属性

1 はじめに

現在，ビッグデータの利用は官民共に盛んである．そうしたデータの利用分野の一つとして機械学習があり，機械学習の利活用もまた盛んである．機械学習を行うためには，空値の補填や値に一貫性をもたせるなどのデータ整備が重要となる．そのため，Data Wrangling として，機械学習に向けたデータクリーニングが着目されている [1]．また，そのような前処理としてのデータ整備に対して，データ利用における作業の 80 %の時間をデータユーザは費やしているという研究も存在している [2]．本研究では，科学メタデータなどの機械的な修正が困難なデータに対し，データ管理者によるインタラクティブな操作によってデータクリーニングを行う方式を提案する．本研究で想定している科学メタデータは，科学データのデータセットに対して，提供者やそのデータが属するカテゴリ，提供された日時といった情報を記載するものであり，データ分析に対して，重要な情報を提供している．しかし，メタデータに自由入力の項目がある場合，様々な表記方法が考えられるため，辞書による入力候補のサジェストなどがない場合は，同じ値が入る項目に対して複数の表記が存在する状況が発生することが多い．そのような場合はメタデータの検索性や利便性が落ちてしまい，利活用を妨げることになる．このような同じ値が入る項目に対して複数の表記が存在していたり，入力の誤りがある場合を，データの不整合と呼ぶこととする．特に，自由入力が多い場合で，かつ，取り扱いに専門的な知識を要する科学データにおいては，全てのメタデータ項目に対して辞書を用意することは困難であり，不整合なデータを生じる可能性が高い．

データの不整合があるようなメタデータの例を図 1 に示す．この例では一つの組が一つのデータに対応しており，それぞれの組に t_1 から t_4 の ID を割り当てている．カテゴリの列はそのデータの属するカテゴリを，提供者の列はそのデータの提供者の名前を，作成機関の列はデータを作成した機関の名前を，収録年はそのデータが収録された年を表している．図 1 の例では，提供者の欄について，t_1 とそれ以外の組とで姓名の順番が異なっており，不整合が起きている．こうした姓名の順番の判断などは辞書を用いるなどといった機械的な判断が難しい．また，機関の略称として “KU” が用いられているが，その略称に対応する機関名は “Kyoto univ.” と “Kobe univ.” の 2 種類が表の中に存在しており，これもまた機械的な判断が難しい．本研究では，こうした機械的に判断することが難しい不整合に対応するため，人間，特にデータ管理者をユーザとして想定し，ユーザがデータの不整合に対して修正方針を判断し，修正していくことを補助するデータクリーニング方式を提案する．

	カテゴリ	提供者	作成機関	収録年
t_1:	ocean	Omori Hiroki	KU	2011
t_2:	forest	Hiroki OMORI	Kyoto univ.	2016
t_3:	soils	Taro YAMADA	Kobe univ.	2017
t_4:	ocean	Hiroki OMORI	Kobe univ.	2018

図 1: 不整合なメタデータの例

本論文では関係データに対するデータクリーニング方式を議論する．関係データベースを扱う利点として，修正したい箇所をビューで指し示すことが可能である点が挙げられる．修正したい箇所をビューで指し示すことができれば，双方向変換技術を用いて，容易にビューから元のデータを修正することが可能である [3]．

本研究は，ユーザが，データマーキング，データ修正，ビュー遷移という 3 つのユーザ操作を使い分けて，不整合なデータを含むビューを効率良く順次得ていき，不整合と判断したデータを修正していくというインタラクティブな方式を提案する．さ

らに，最初に不整合なデータを含みそうなビューをユーザに提示する手法についてや，データを修正していく中で知見を得ることができ，それをどのようにして活用するかということについても議論を行った．

本論文の構成を以下に示す．第 2 節では提案するデータクリーニング方式がどのようなものであるかを説明する．第 3 節では実際のメタデータを観察し，本研究が対象とするような問題の実例を観察する．第 4 節では本論文と関連している研究を紹介する．第 5 節ではユーザがデータを修正していく中で得られた知見の利活用方法や，最初にユーザが不整合なデータを発見するための足がかりとしての，初期のビューを自動で発見する手法の検討についての議論を行う．第 6 節では，本論文のまとめを述べ，今後の課題について議論する．

2 提案するデータクリーニング方式

関係データベースに格納されたデータを対象に，ユーザとのインタラクションを通じて，不整合なデータの修正を行っていくデータクリーニング方式を提案する．本研究における不整合なデータとは，同じ対象を指しているのに，値が異なるデータや，そもそも入力値に誤りがあるようなデータを指している．

本研究は，ユーザが，データマーキング，データ修正，ビュー遷移という 3 つのユーザ操作を使い分けて，不整合なデータを含むビューを効率良く順次得ていき，不整合と判断したデータを修正していく方式を提案している．本論文では，不整合なデータを含むビューを得ていくために，手がかりとして，最初に不整合なデータを含むビューをユーザが，初期ビューとして与えることを想定している．図 2a は，ユーザが与えた初期ビューの例である．図 1 と同様に，各組が対応するデータの説明を行っている．各属性の意味も図 1 と同様である．図 2a の例では，ユーザが “カテゴリ=ocean” の条件節で指定されるビューに不整合な値が含まれていると判断し，そのビューを初期入力として与えている状況を想定している．

2.1 ユーザ操作の概要

本論文は，システムとユーザとのインタラクションを通してビューに含まれる不整合なデータを修正していく方式を提案している．そのため，ユーザがシステムから提示されたビューを見ながら取ることが可能ないくつかの操作を考えており，それらをユーザ操作と呼ぶこととする．ユーザ操作として，データマーキング，データ修正，ビュー遷移の 3 つを考えた．それらは次の通りである．

（1） データマーキング: ユーザが不整合だと考えた値に印をつける操作

（2） データ修正: 印をつけた値をユーザが望む値に修正する操作

（3） ビュー遷移: ユーザが不整合なデータを探しやすいように適切な箇所を含めた適切なサイズのビューを得る操作

さらにビュー遷移として 3 種類の遷移方法を考えた．それらは次の通りである．

（1） ビュー拡大: 現在見ているビューの条件節を 1 つ削減し，ビューのサイズを拡大する操作

（2） ビュー縮小: 現在見ているビューの条件節を 1 つ追加し，ビューのサイズを縮小する操作

（3） ビュー移動: 現在見ているビューから，データマーキングでつけられた印などを考慮して，新しい箇所を示すビューを得る操作

ビュー遷移を行って適切なサイズのビューを得ながら，その中で不整合とユーザが判断した値をデータマーキングし，それらの値に対してデータ修正を実施するといった流れを想定している．

2.2 操作の流れ

実際にどのように操作を組み合わせてデータを修正したり遷移したりするのかを図 2a の例を用いて示す．

2.2.1 データ修正の流れ

ユーザがデータを修正するまでの流れを説明する．まずユーザが不整合だと考えた値にデータマーキングで印をつける．図 2a の例だと，作成機関の属性の値が，“Kyoto univ.” と書かれていてほしいところが “KU” になっているといった状況を想定している．そのため，作成機関が “KU” となっている上から 2 つの組，“ID=100” と “ID=68” である組の “KU” の値にユーザによってデータマーキング操作により印が付与される状態を想定する．ここで，ユーザがデータの修正を目的にしている場合，作成機関が “KU” となっている組のみに絞り，より集中して目を通したいような状況が考えられる．その場合は，ユーザがこのビューを指定する条件節である “カテゴリ=ocean” に，新しく “作成機関=KU” という条件を追加し，ビュー縮小を行う．これにより得た “カテゴリ= ocean AND 作成機関=KU” の条件節が示すビューは，さきほどのビューより範囲が狭くなっている．この操作は，“KU” は “Kyoto univ.” に置き換えるべき値かどうかをユーザが判断しやすくするための操作である．この操作の結果，図 2b のビューが得られたとする．ここでは，“KU” は “Kyoto univ.” に置き換えるべきだと判断し，ユーザは，“カテゴリ=ocean AND 作成機関=KU” のビューの範囲内で，作成機関の値が “KU” となっている箇所を “Kyoto univ.” に変更するようなデータ修正の操作を行う．最後に，ビューを示す条件節から “作成機関=KU” を削除，“カテゴリ=ocean” の条件節が示すビューへとビュー拡大を行い，一部の “KU” が “Kyoto univ.” へと修正された状態の図 2a のビューへと帰ってくる．この一連の修正により，作成機関が “KU” となっているところを，“Kyoto univ.” へと修正したという知識が得られる．この知識は，第 5 節にて利用方法を議論する．

2.2.2 ビュー遷移の流れ

ユーザが新しい不整合なデータを見つけていくために行う一連の操作を説明する．この操作は不整合なデータを発見するために，ユーザが見通しやすいビューを得ることを目的としている．まず，図 2a の段階に戻る．この表に対するデータマーキングとして，先程と同様に，上から 2 つの組の “KU” の値に対して，ユーザから印が付与されたことを想定する．ここでユーザ

ID	カテゴリ	提供者	作成機関	連絡機関	収録年
100	ocean	OMORI	KU	Yoshikawa lab	2018
68	ocean	OMORI	KU	Yoshikawa lab	2015
59	ocean	YAMADA	Kyoto univ.	kyoto univ.	2016
58	ocean	SUZUKI	Kyoto univ.	Kyoto univ.	2016
43	ocean	SUZUKI	Kyoto univ.	Kyoto univ.	2014
...	...	...	...	...	...

(a) 初期ビューの例: カテゴリ=ocean

ID	カテゴリ	提供者	作成機関	連絡機関	収録年
100	ocean	OMORI	KU	Yoshikawa lab	2018
68	ocean	OMORI	KU	Yoshikawa lab	2015
57	ocean	OMORI	KU	Yohsikawa lab	2016
31	ocean	OMORI	KU	KU	2013
45	ocean	OMORI	KU	KU	2014
...	...	...	...	...	...

(b) ビュー縮小の例: カテゴリ=ocean AND 作成機関=KU

ID	カテゴリ	提供者	作成機関	連絡機関	収録年
100	ocean	OMORI	KU	Yoshikawa lab	2018
68	ocean	OMORI	KU	Yoshikawa lab	2015
57	ocean	OMORI	KU	Yohsikawa lab	2016
34	ocean	OMORI	OU	OU	2013
49	ocean	OMORI	OU	OU	2014
...	...	...	...	...	...

(c) ビュー縮小の例: カテゴリ=ocean AND 提供者=OMORI

ID	カテゴリ	提供者	作成機関	連絡機関	収録年
100	ocean	OMORI	KU	Yoshikawa lab	2018
68	ocean	OMORI	KU	Yoshikawa lab	2015
56	forest	OMORI	KU	Yohsikawa lab	2016
33	climate	OMORI	OU	OU	2013
47	soil	OMORI	KU	Yoshikawa lab	2014
...	...	...	...	...	...

(d) ビュー拡大の例: 提供者=OMORI

図 2: ビュー遷移の例

は，不整合な値を持つ組を指す条件として，“カテゴリ=ocean AND 提供者=OMORI”を発見したとする．ユーザは，この条件を参考に，現在のビューを示す “カテゴリ=ocean”に新しく “提供者=OMORI”の条件を追加することにより，ビュー縮小を行う．その結果，図 2c のような “カテゴリ=ocean AND 提供者=OMORI”のビューをユーザは新たに得る．このビューを見た結果，OMORI という提供者は，“Kyoto univ.”を “KU”という略記をしていた他に，他にも “OU”という略記を用いているため，このビューの範囲外でも同様のデータ不整合が発生している可能性があることが観察できる．実際のデータでも，このような提供者がある箇所で他と異なる表記を行っていた場合，別の箇所でも同様の異なる表記を用いているといった状況が観察された．そのことについては，第 3 節で議論する．この例では，提供者 OMORI が異なるカテゴリでも同様な略記を用いている可能性を考え，現在のビューを示す条件である “カテゴリ=ocean AND 提供者=OMORI”から，“カテゴリ=ocean”を削除することによりビュー拡大を行う．その結果，図 2d のような “提供者=OMORI”のビューをユーザは新たに得る．こうして，他のカテゴリにおいても，“提供者=OMORI”は他と異なる表記を用いていることが確認できた．

また，この一連のビュー遷移の操作は，ビュー移動の 1 つの操作にまとめることが可能である．再び，図 2a の段階に戻る．先程と同様に，この表にデータマーキングが行われ，ユーザが不整合な値を持つ組を指す条件として，“カテゴリ=ocean AND 提供者=OMORI”を発見したとする．この条件から，ユーザは，“提供者=OMORI”が他のカテゴリでも不整合な値を入力していると推測し，直接，“提供者=OMORI”のビューを得ることが可能である．このような不整合な値を持つ組を指す条件から直接別の箇所を指すビューを導く操作を，ビュー移動とする．このビュー移動という操作は，先程の説明から，ビュー拡大とビュー縮小により再現することが可能であると分かるが，ユーザの利便性を考え，個別に定義した．なお，実用的にはユーザが不整合な値を持つ組を指す条件を発見するのは困難な場合もあると思われるが，システムが自動的にこのような条件を発見する手法を考案することは，今後の研究課題としている．

3 実 例 観 察

本研究では，データ統合・解析システム DIAS (Data Integration and Analysis System)[1]に提供されたデータに付属しているメタデータの一部を用いて，実際に本研究が提案する方式が有用となる事例の観察を行った．DIAS のメタデータは本来 XML 形式であるが，それを簡易的に 12 属性の関係データへと変換し提案する方式を適用した．一つの行が一つのデータセットに対応しており，今回は 426 データセットのメタデータを観察した．このメタデータは，カテゴリ，制作された日時，提供されたデータセットの作成者，その所属機関，メタデータの著者やその所属機関を属性として持つ．DIAS は観測によって得られた地球各地での多様な観測データを収集しており，それに付随しているメタデータも多様であるため，科学データを対象にした観察として有用である．上記データの属性のうち，そのデータがどのようなカテゴリのデータであるかを示しているカテゴリの属性，そのデータのドキュメントを書いた著者名を表す著者名の属性，その著者の所属機関名を表す所属機関の属性，そのデータセット自体を作成した人の名前を表す作成者名の属性，その作成者の所属機関名を表す作成機関の属性，そして，データについて問い合わせを行う連絡先の機関名を表す連絡機関の属性に着目してデータを観察した．第 2 節で述べた例と同様に，最初に初期ビューとして，“カテゴリ=ocean”を与える．その中で更に，連絡機関と著者名に着目した結果が図 3 である．イタリック体で強調した 1，2 番目の組の

1 : http://www.diasjp.net

連絡機関	著者名
JAMSTEC/DrC	Hiromichi Igarashi
JAMSTEC/DrC	Hiromichi Igarashi
Center for Global Envir...	Shin-ichiro Nakaoka
JAMSTEC/RIGC	Hiromichi Igarashi
NULL	Japan coast guard hydrogr...
Atmoshpere and Ocean...	Sachihiko Itoh
Japan Agency for Marin...	JAMSTEC-CEIST
Center for Global Envir...	Nojiri, Yukihiro
Center for Global Envir...	Nojiri, Yukihiro
DrC/JAMSTEC	Hiromichi Igarashi
...	...

図 3: 実例における初期ビューの一部

著者名	所属機関	作成者名	作成機関
Hiromichi Igarashi	JAMSTEC/DrC	Sugiura, Nozomi, Dr.	**JAMSTEC/DRC**
Hiromichi Igarashi	JAMSTEC/DrC	Dr. Nozomi Sugiura	**JAMSTEC/DRC**
Hiromichi Igarashi	**Japan Agency for Marine-Earth...**	Kazuo Umezawa	Japan Aerospace Exploration...
Hiromichi Igarashi	**Japan Agency for Marine-Earth...**	Kazuo Umezawa	Japan Aerospace Exploration...
Hiromichi Igarashi	**Japan Agency for Marine-Earth...**	Kazuo Umezawa	Japan Aerospace Exploration...
Hiromichi Igarashi	**Japan Agency for Marine-Earth...**	Kazuo Umezawa	Japan Aerospace Exploration...
Hiromichi Igarashi	**Japan Agency for Marine-Earth...**	Kazuo Umezawa	Japan Aerospace Exploration...
Hiromichi Igarashi	JAMSTEC/DrC	Hiromichi Igarashi	JAMSTEC/DrC
Hiromichi Igarashi	JAMSTEC/DrC	Hiroshi Kawamura	Center for Atmospheric and...
Hiromichi Igarashi	JAMSTEC/DrC	Sugiura, Nozomi, Dr.	**JAMSTEC/DRC**
Hiromichi Igarashi	**DrC/JAMSTEC**	Remote Sensing Systems	NULL
...	...	...	...

図 4: 実例におけるビュー移動後のビュー

ように，連絡機関の値について，このビューの範囲内全体では “JAMSTEC/DrC” という書き方が主流である．しかし，太字で強調した最後の組では，連絡機関の値は “DrC/JAMSTEC” となっている．よって，主流の表記である “JAMSTEC/DrC” に合わせて “DrC/JAMSTEC” を変更したいといった状況である．そこで，第 2 節で用いた例のように著者名に着目し，同じ著者が同様の表記をしていないか他のカテゴリも確認するため，現在の “カテゴリ=ocean” という条件で示されるビューから “著者名=Hiromichi Igarashi” で示されるビューへとビュー移動を行う．その結果が，図 4 である．

属性は著者名，所属機関，作成者名，作成機関に着目した．このビューでは，太字で強調した箇所のように，所属機関が “DrC/JAMSTEC” や，JAMSTEC の正式名称である “Japan Agency for Marine-Earth Science and Technology” といった値となっている組が観察された．また，作成機関が “DRC/JAMSTEC” となっている箇所が観察された．この中でも所属機関の値が “Japan Agency for Marine-Earth Science and Technology” となっている組は，作成機関の値が，“Japan Aerospace eXploration Agency” となっており，そちらの表記に合わせた可能性がある．このように一見，主流の表記とずれている場合でも，データ作成者の意図が存在している場合があるため，人間が表記を修正するかどうか検討する必要性がある例となっている．また，これらの表記の違いは，“DrC/JAMSTEC” のデータ不整合を発見した属性である連絡機関と異なる属性で発見されているため，同じ属性で同じような表記を探すやり方では発見が困難となる例となっている．上記の 2 点のような本研究が有用であるような状況が，実際の例の中で観測された．

4 関 連 研 究

本論文ではユーザとシステムがインタラクティブにデータクリーニングを行う方式を採用している．そのことの是非については， Volkovs らの研究 [4] で論じられている．従来のデータクリーニングは，データベースについての一貫性制約を考慮する静的なデータベースを対象としていた．しかし，近年では，一貫性制約も時々刻々と変化する動的な環境が増えていることから，値が不整合であるのか，一貫性制約がアップデートされたのかが不明瞭であり，従来の制約とデータのみを考慮するバッチ処理的なデータクリーニングには限界が生じている．そこで Volkovs らの研究では，システムとインタラクティブにやり取りしながらユーザに操作させることで，ユーザの判断を取り込む手法が提案されている．

同様の，ユーザとシステムがインタラクティブにデータクリーニングを行う研究としては， He らの研究 [5] がある．この研究は，まずユーザがテーブルに格納されているデータを調べ，データに対しての修正であるリペアを作成する．そのリペ

アに対して，システムが更新ルールとして用いるために，SQL 更新クエリの集合を生成する．こうして作成された SQL 更新クエリに対して，ユーザが適切であるかどうかをチェックし，もし正しければ，そのクエリを用いてより多くのデータを修正するというワークフローとなっている．ユーザがシステムとインタラクティブに修正を行うという点は本論文と類似しているが，本論文はユーザが効率よく修正できるよう，ユーザへ不整合な値がある可能性が高いビューを提示するという点が異なっている．

このようにビューのようなデータのサブセットをユーザに提示し，データクリーニングを行う研究としては， Rahman らの研究 [6] がある．この研究は，不整合な値をクリーニングすることではなく，意図的に報告がなされていない欠損値を埋めることを目的としている．ここでいう意図的に報告がなされてない欠損値とは，ミスやエラーによりランダムに欠損してしまった値ではなく，専門家からすると自明であるため，意図的に入力が放棄された値である．このような値は，専門家の知見なしに単純な分析などを用いてクリーニングすると精度が非常に悪いものとなる．そのため，対象のデータベースの専門家に効率よく欠損値を含むデータベースのサブセットを提示し，欠損値を埋めてもらうという方針を取っている．また，ユーザがデータベースのサブセットを埋めた結果から欠損値を埋めるためのルールを推論し，更に，属性間の階層構造を考慮して，より多くの欠損値を埋めるようなルールもユーザに提案するシステムを考案している．このような専門家の知見を用いて，データの整備の精度を向上させる方針は本論文と共通している．本論文は，不整合な値の種類を欠損値に限定することなく，ユーザが不整合だと考えた値を対象にし，効率よくビューを提示していく点で異なっている．加えて，サブセットではなくビューをユーザに編集してもらうことにより，不整合な値を含むタプルを指す条件を探し，その条件を用いて更に不整合のありそうなタプルを含むビューを提示していく手法も，Rahman らの研究と異なる点である．

5 議　　論

5.1 得られた知識の利活用方法

第 2 節で述べた通り，本研究のデータクリーニング方式を使用すると，どのような不整合なデータをどのように修正したかについての履歴や，どのようなビューを見ている際に不整合なデータに修正が行われたかについての履歴が得られる．どのような不整合なデータをどのように修正したかについての履歴は，履歴にある不整合なデータとなりうるような入力が新しくなされた時に，それに対応する修正後の入力をシステムがサジェストするといったような利用方法が可能である．その他にも，辞書の作成が進んでいないような専門性の高い分野の同義語を，同じ対象を指している言葉として検索出来るようにするために利用することも可能であると思われる．更に，このような履歴は，Yakout らの研究 [7] のように，機械学習を用いてユーザの望むような修正を予測する手法への利用も可能であると思われる．

5.2 初期ビューの発見手法の検討

現在の提案しているデータクリーニング方式では，初期のビューはユーザが入力として与えることを想定しているが，適切な初期ビューをシステムが自動で発見することでユーザを補助することは有用であると考えられる．データに不整合があるような箇所を発見する手法として，関数従属性を用いる手法が多く研究されている [8–10]．これらの手法は，関数従属性が成立するような組を探すために，パターンマイニングの考えを応用し，confidence と support の高い，各属性の等価クラスを要素に持つアイテム集合を発見することによって，関数従属性を成立させるために除くべき最小の組集合を発見するやり方が主流である [11, 12]．しかし，関数従属性は，テーブル単位で属性ごとに与えられるため，第 2 節の例における所属機関や第 3 節の実例における連絡機関といった同じ人間に複数の所属があるが，しかしその中で不整合なデータを発見したいといった状況で使うには不適当である．

そこで，関数従属性を発展させた概念である，conditional functional dependency (CFD) の考えを適用することを検討している．CFD も関数従属性と同様，多くのデータクリーニングツールに利用されている [13, 14]．CFD は関数従属性にパターンタプルという，関数従属性に現れる属性の値のパターンを指定するタプルを組み合わせたものであり，パターンタプルのパターンを満たした組だけが関数従属性が成立すべき組だとみなす，条件付きで成立する関数従属性のような制約である [15, 16]．そのため，関数従属性と異なり，同じ人物が複数の所属に属している場合でも，CFD ならば制約を考えることが可能である．よって関数従属性より CFD のほうが本研究に適していると考えている．しかし，第 2 節で取り扱ったような，同じ人物が一貫して不整合なデータ入力を行っているような場合では，CFD で不整合を捉えることはできない．このような検出することが困難な不整合も，本研究の方式により修正することができる可能性があると考えている．

6 おわりに

本研究は，同じ値が入る項目に対して複数の表記が存在していたり，入力の誤りがある場合のデータを不整合なデータと呼び，効率よく不整合なデータを含むビューをユーザに提示していくことで，データを修正していくデータクリーニング方式を提案した．また，実際のデータを観察し，提案するデータクリーニング方式が有用となる状況を確認した．そして，提案するデータクリーニング方式を実行していく中で得られた知識をどのように活用するか，また，提案方式を適用する際に，本論文ではユーザ入力として与えられる初期ビューを自動で発見するための方針について議論を行った．

今後の課題としては，第 5 節でも議論したように，初期ビューを自動で発見する手法を考案することが挙げられる．同様に，第 5 節で議論した，提案手法を実行する中で得られた知識の活

用方法も更に考察したい．また，実際のデータにはノイズが多く含まれているが，その中でどのようにデータマーキングがなされた組だけを指す条件を発見するのかという課題も存在している．更に，データベースには通常多くの属性が存在するが，属性数も組数と同じ様な発想で見やすいサイズに限定すべきかを検討することも課題として挙げられる．

謝　　辞

本研究の一部は JSPS 科研費 JP17H06099, JP18H04093, JP18K11315 の助成を受けたものです．

文　　献

[1] Sean Kandel, Jeffrey Heer, Catherine Plaisant, Jessie Kennedy, Frank van Ham, Nathalie Henry Riche, Chris Weaver, Bongshin Lee, Dominique Brodbeck, and Paolo Buono. Research directions in data wrangling: Visualizations and transformations for usable and credible data. *Information Visualization*, 10(4):271–288, 2011.

[2] Tamraparni Dasu and Theodore Johnson. *Exploratory data mining and data cleaning*, volume 479. John Wiley & Sons, 2003.

[3] Yasuhito Asano, Soichiro Hidaka, Zhenjiang Hu, Yasunori Ishihara, Hiroyuki Kato, Hsiang-Shang Ko, Keisuke Nakano, Makoto Onizuka, Yuya Sasaki, Toshiyuki Shimizu, et al. A view-based programmable architecture for controlling and integrating decentralized data. *arXiv preprint arXiv:1803.06674*, 2018.

[4] Maksims Volkovs, Fei Chiang, Jaroslaw Szlichta, and Renée J Miller. Continuous data cleaning. In *Data Engineering (ICDE), 2014 IEEE 30th International Conference on*, pages 244–255. IEEE, 2014.

[5] Jian He, Enzo Veltri, Donatello Santoro, Guoliang Li, Giansalvatore Mecca, Paolo Papotti, and Nan Tang. Interactive and deterministic data cleaning. In *Proceedings of the 2016 International Conference on Management of Data*, pages 893–907. ACM, 2016.

[6] Protiva Rahman, Courtney Hebert, and Arnab Nandi. Icarus: minimizing human effort in iterative data completion. *Proceedings of the VLDB Endowment*, 11(13):2263–2276, 2018.

[7] Mohamed Yakout, Ahmed K Elmagarmid, Jennifer Neville, Mourad Ouzzani, and Ihab F Ilyas. Guided data repair. *Proceedings of the VLDB Endowment*, 4(5):279–289, 2011.

[8] Marcelo Arenas, Leopoldo Bertossi, and Jan Chomicki. Consistent query answers in inconsistent databases. In *Proceedings of the eighteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 68–79. ACM, 1999.

[9] Jan Chomicki and Jerzy Marcinkowski. Minimal-change integrity maintenance using tuple deletions. *Information and Computation*, 197(1-2):90–121, 2005.

[10] Jef Wijsen. Database repairing using updates. *ACM Transactions on Database Systems (TODS)*, 30(3):722–768, 2005.

[11] Yka Huhtala, Juha Kärkkäinen, Pasi Porkka, and Hannu Toivonen. Tane: An efficient algorithm for discovering functional and approximate dependencies. *The computer journal*, 42(2):100–111, 1999.

[12] Catharine Wyss, Chris Giannella, and Edward Robertson. Fastfds: A heuristic-driven, depth-first algorithm for mining functional dependencies from relation instances extended abstract. In *International Conference on Data Warehousing and Knowledge Discovery*, pages 101–110. Springer, 2001.

[13] Lukasz Golab, Howard Karloff, Flip Korn, Divesh Srivastava, and Bei Yu. On generating near-optimal tableaux for conditional functional dependencies. *Proceedings of the VLDB Endowment*, 1(1):376–390, 2008.

[14] Wenfei Fan, Floris Geerts, and Xibei Jia. Semandaq: a data quality system based on conditional functional dependencies. *Proceedings of the VLDB Endowment*, 1(2):1460–1463, 2008.

[15] Philip Bohannon, Wenfei Fan, Floris Geerts, Xibei Jia, and Anastasios Kementsietsidis. Conditional functional dependencies for data cleaning. In *Proceedings of the 23rd International Conference on Data Engineering*, pages 746–755. IEEE, 2007.

[16] Wenfei Fan, Floris Geerts, Jianzhong Li, and Ming Xiong. Discovering conditional functional dependencies. *IEEE Transactions on Knowledge and Data Engineering*, 23(5):683–698, 2011.