学術文献におけるテキスト分析タスクのためのデータセット自動構築システムの開発

Li Jinghong[*1], 太田 光一[*1], 谷 文[*1], 長谷川 忍[*1]
*1 北陸先端科学技術大学院大学

Development of an automatic dataset construction system for text analysis tasks in academic literature

Jinghong Li [*1], Koichi Ota[*1], Wen Gu[*1], Shinobu Hasegawa[*1]
*1 Japan Advanced Institute of Science and Technology

In machine learning tasks for text analysis of academic literature, collection and preprocessing of literature files are essential. However, it is not easy to process a large amount of data manually. Therefore, this study focuses on web-scraping of academic literature from JALC (Japan Link Center) and develops a system that automatically builds a dataset for machine learning tasks.

キーワード: テキストマイニング，ウェブスクレイピング， PDF 構造解析，文章分割，データセット自動構築

1. はじめに

インターネットの普及により膨大な情報が溢れている現在，大量のテキストデータから重要な情報を把握することが我々にとって重要なスキルとなっている．その一例として，本研究では研究初学者の大学院生を対象とする．研究初学者が研究分野の全体像や研究の方向性を把握するために，出版されている学術論文を読み・活用することが不可欠である(1)．しかし，電子化された大量の学術論文が日々出版・公開される今日では，それら全てに目を通すことさえもままならない．また, 研究の方向性が定まっていない研究者にとって, 学術論文から研究目的や研究方法，評価，結果，考察などに代表される論文の構成要素を効率よく把握することは困難である．さらに，異なる著者による様々な構造を持つ論文から得られた多様な情報を適切に構造化することは研究初心者にとって複雑なタスクであり, 容易ではない．これらの問題を解決するために，学術論文におけるテキスト分析が使用される．筆者らは機械学習を用いた自動要約システムを提案した(1)．しかしながら，小規模なデータだけでトレーニングすると過学習になりやすい．汎用性を向上するためには，大量の文章データやメタデータを機械学習モジュールに入力することが不可欠である．そのため，効率的な論文関連データの収集手段としてウェブスクレイピング技術がよく使用されている．

日本語を対象としたスクレイピング技術として，中智らはWeb上から自動で表データを抽出するスクレイピングWeb APIを開発した(2)．石井らは，医療関連のデータや情報を収集するためにウェブスクレイピングを使用した(3)．しかし，上記の研究では一般にPDFで公開されている学術論文内に含まれるデータは扱われていなかった．

本研究では，JALC(Japan link center)の一般向けデータ提供サービス利用規約に従い(4)，JALCからのスクレイピングを行う．JaLC に登録されているプレフィックスリスト，DOIリスト，および書誌データや URI, 引用情報等(5)を取得できるサービス ー「JaLC REST API」を利用し，日本語文献ファイルを収集する．また，本研究の文献ファイルデータ収集については，著作権法改正（2021年1月施行）により，著作権法第30条の4(6,7)

に従い，収集した著作物に対する人工知能の開発に関する情報解析の為利用する．著作物に表現された思想又は感情の享受を目的としないと行為であると考えられる. こうした収集した日本語文献ファイルを対象に，論文構造を含むテキスト情報を抽出し，論文データマイニングを行うためのデータセット自動収集システムを開発する．これにより，学術論文を対象としたデータマイニングや機械学習のための前処理に費やす時間が大幅に短縮できると期待される.

2. 関連研究

2.1 学術論文を対象としたウェブスクレイピング

Wahaj らは英語対応の科学ジャーナルからデータを抽出するためのスクレイピングアプリケーションを開発した[8]．日本語の学術論文を対象とするウェブスクレイピングの既存研究として，久保らは J-STAGE（国立研究開発法人科学技術振興機構 (JST) が運営する電子ジャーナルプラットフォーム）に対し，Web スクレイピングを通じて論文の書誌情報や著者情報を機械的に取得することを検討した[9]．スクレイピングの対象は Web ページに書かれたメタ情報のみであり，論文内容は扱われていない.

2.2 文献ファイルに対するテキストマイニング

文献ファイルに対するテキストマイニングの関連研究として，Clark らは，広く適用可能な英文文献ファイルを対象とするテキスト分類を試行し，図領域，セクション，タイトルの検出のためのクラスタリングメカニズムを開発した[10]. Yang らは研究者が論文データの関係を理解し，迅速に洞察を得ることを支援することを目的として，材料科学論文から手順情報（レシピステップ），図，表を抽出し，機械学習に基づき検索および可視化の機能を持つ手順型情報抽出・知識管理システム（PIEKM）を開発した[11].

しかし上記の関連研究では，日本語を対象としておらず，構造が異なる日本語文献に適用するのが困難である.

2.3 本研究の位置づけ

(1)データ収集の効率化：本研究では，特定の学術論文誌を初期の対象としてウェブスクレイピングを通じて日本語の学術論文 PDF ファイルを効率的に収集する方法を提案する．さらに，研究分野の特徴に応じたメタデータの整備を行う.

(2)テキストの前処理部分：PDF から取得されるテキストデータにはノンブルや不要な改行などのノイズや図表などのデータが含まれており，そのままでは後の解析が難しい．本研究では，学術論文特有の特徴を反映した前処理を行うことで，論文構造分析の精度向上を目指す.

(3) 階層化された論文構造データセット：本研究では，論文の章節構造を反映し，文章単位まで分割されたデータセットを自動で生成する．こうして生成されたデータセットは，機械学習のタスクで使いやすいと考えられる.

3. データセット自動構築システム

本研究では，学術論文を対象とした機械学習タスクで利用可能なデータセットを自動的に構築するシステムを開発する．システムの全体の構成を図 1 に示す.

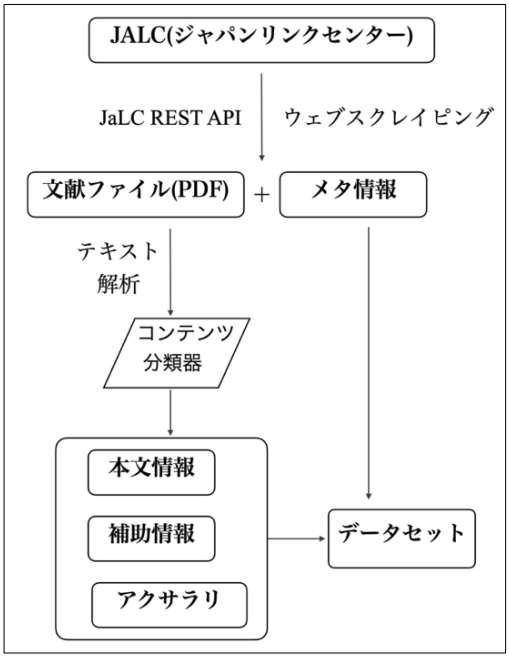


図 1 システム全体像

データセット自動構築の流れとしては，まず JALC（日本リンクセンター）と呼ばれる学術文献データベースから，論文誌の文献収集に特化したウェブスクレイピングを適用して，論文誌毎の文献ファイルとその

メタデータを一括で取得する．そして，データセットを構築するために，表 1 で定義した文献内部のコンテンツの分類抽出を目指す．具体的には，PDF 解析ツールによって得られた各テキストブロック領域に対応するコンテンツ内容を自動的に判断する．最後に，ウェブスクレイピングで収集したメタ情報と文献内部コンテンツを組み合わせて論文構造を反映したデータセットを構築する．

表 1 学術論文内部コンテンツ定義

種類	明細
本文情報	本文文章群
補助情報	章節，図，表，数式，引用マーク
アクセサリ	タイトル, アブストラクト，キーワード，脚注

3.1 論文誌の文献収集に特化したウェブスクレイピング

学術論文のデータにアクセスするため，DOI 識別子が使われる．DOI の機能はシンプルで，個別のコンテンツに割り振られた ID (DOI)とその所在 URL 情報をペアで保管し，DOI への問い合わせに対して所在 URL を返すというものである[12]．つまり，DOI 情報さえあれば，Web ページ操作の代わりに DOI リンクにアクセスして，学術論文データが含まれるページに遷移することができる．そのため，DOI 番号や DOI リンクの取得を中心とするウェブスクレイピング技術が使用できる．本研究では，ウェブスクレイピングの対象となるウェブサイトは JALC とする．JALC とは，電子化された学術論文，書籍，論文付随情報，研究データなどに DOI を登録し，コンテンツの所在情報(URL)等とともに管理している論文データベースである．そのデータベースに直接アクセスするために，2021 年末に公開された JALC REST API[13,14]を利用し，サーバーへリクエストメッセージを発信して，そのレスポンスから検索したい情報を一括取得することができる．本研究で主に使用した情報は文献ファイルにアクセスできる DOI リンクや論文のメタデータであり，スクレイピング部分の全体像を図 2 に示す．次は各部分におけるスクレピングの方法を紹介する．

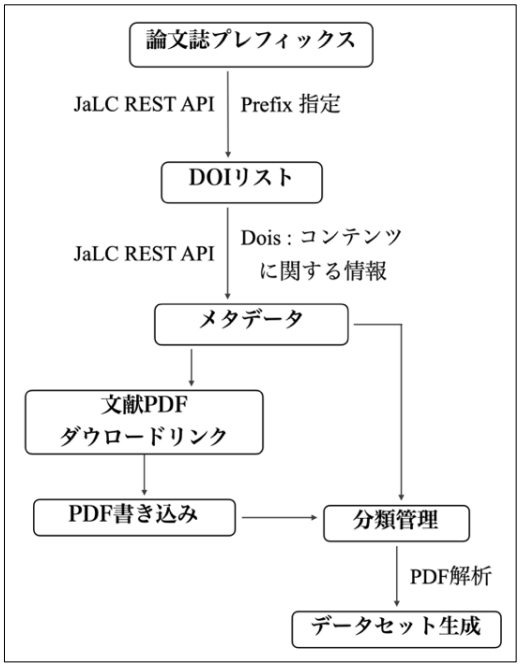


図 2 スクレイピングの流れ

3.1.1 DOI リストの取得（プレフィックス）

図 3 に示した DOI の構造では，赤い枠に囲まれた部分はプレフィックスと呼ばれ，論文誌ごとに特定のプレフィックスが付与される．

図 3 DOI 構造やプレフィックス

本研究で使用した JSiSE 論文誌のプレフィックスは 10.14926 である．プレフィックスが指定されると，サーバーへのリクエスト：

```
https://api.japanlinkcenter.org/dois/$10.14926
```

を発信し，json フォーマットの DOI リストが返信される．その DOI リストの解析には，python の「ast」モジュールを用い，テキストを辞書の型(dict)に変換してから，DOI 番号を取得する．

3.1.2 メタデータ情報整合

DOI リストに格納された DOI 番号の値をサーバーへのリクエストとして発信すると，DOI 番号に対応する論文メタ情報がレスポンスされ，詳しい関連情報を取得することができる．図 4 はメタデータの一部である．

{"status":"OK","apiType":"doi","apiVersion":"1.0.0","message":
{"total":1,"rows":1,"totalPages":1,"page":1},"data":
{"siteId":"SI/JST.JSTAGE","content_type":"JA","doi":"10.14926/jsise.38.305","url":"https://doi.org/10.14926/jsise.38.305","ra":"JaLC","prefix":"10.14926","site_name":"J-STAGE","publisher_list":
[{"publisher_name":"Japanese Society for Information and Systems in Education","lang":"en"},
{"publisher_name":"教育システム情報学会","lang":"ja"}],"title_list":[{"lang":"en","title":""Common Story of DX" or "Barriers of DX""},{"lang":"ja","title":"「DXあるある」あるいは「DXの壁」"}],"creator_list":
[{"sequence":"1","type":"person","names":[{"lang":"en","last_name":"Hasegawa","first_name":"Shinobu"},
{"lang":"ja","last_name":"長谷川","first_name":"忍"}],"affiliation_list":[{"affiliation_name":"北陸先端科学技術大学院大学","sequence":"1","lang":"ja"},{"affiliation_name":"Japan Advanced Institute of Science and Technology","sequence":"1","lang":"en"}]}],"publication_date":
{"publication_year":"2021","publication_month":"10","publication_day":"01"},"relation_list":
[{"content":"https://www.jstage.jst.go.jp/article/jsise/38/4/38_380401/_pdf","type":"URL","relation":"fullTextPdf"}],"content_language":"ja","updated_date":"2021-09-30","article_type":"pub","journal_id_list":
[{"journal_id":"1341-4135","type":"ISSN","issn_type":"print"},{"journal_id":"2188-0980","type":"ISSN","issn_type":"online"},{"journal_id":"jsise","type":"JID"}],"journal_title_name_list":
[{"journal_title_name":"Transactions of Japanese Society for Information and Systems in Education","type":"full","lang":"en"},{"journal_title_name":"教育システム情報学会誌","type":"full","lang":"ja"}],"journal_classification":"01","journal_txt_lang":"ja","recorded_year":"2013-2014","volume":"38","issue":"4","first_page":"305","last_page":"306","article_number":"380401","date":"2021-10-01","citation_list":[{"sequence":"1","original_text":"(1)　文部科学省デジタル化推進本部:"文部科学省におけるデジタル化推進プラン",https://www.mext.go.jp/content/20210412-mxt_jyohoka01-000014099_13.pdf(参照2021.8.2)"}]}}

図 4 メタデータ取得例

3.1.3 文献ファイルの書き込み

本節では文献ファイルをローカル環境に保存する手順を説明する．まず，節 3.1.2 の説明の通りに取得した json ファイルを解析する．json ファイルで，属性「url」に登録された値は論文のコンテンツリンクであり，そのリンク先のウェブページに対して，HTML 文字列からデータを抽出し解析する Python 言語パッケージ Beautiful soup[15]を利用して HTML 解析を行う．次に，HTML ファイル内のテキストから，属性「class_paper_pdf」の値を取得して，文献 PDF ファイルのページにアクセスする．最後に，その文献 PDF ファイルをローカル環境に保存すれば，スクレピング作業を完了する．なお，JALC サーバーに負担をかけないよう，一定の時間間隔を設定してデータを収集する．

3.2 文献ファイルの前処理

本章では，ルールベースに基づくテキストブロック解析や論文コンテンツ分類により洗練されたデータセットを構築する方法を紹介する．その全体像は図 5 の通りである．

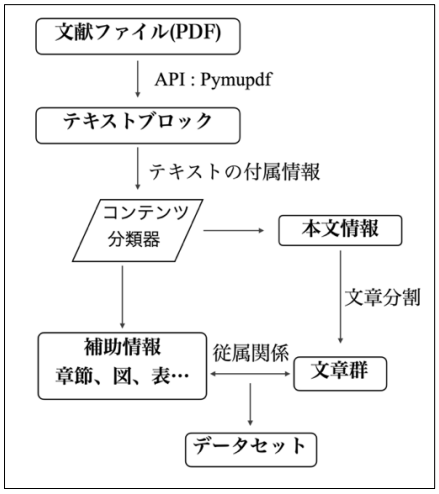


図 5 文献ファイル前処理の流れ

3.2.1 論文構造解析

Python では外部ライブラリ「pymupdf」[16]を使用することで，PDF 操作を自動化することができる．また，学術論文の RAW 構造を見ると，行間隔や列間隔の特徴が把握でき，PDF ファイルのテキストを複数のサブテキストブロックに分けることができる．各テキストブロックには特定のコンテンツが含まれ，それらの組み合わせがページのレイアウトになる．「pymupdf」の「get_text_block」メソッドを使用して，各ページのテキストブロックを抽出することができる．

3.2.2 正規表現に基づく領域分割

学術論文の内部コンテンツ領域は，基本情報領域，本文領域，参考文献領域，付録領域に大きく分類することができる．本稿では，付録以外の領域に正規表現で領域の先頭部分の文字列をマッチすることで領域の分割を行う．例えば，章節「はじめに」との類似表現は「まえがき」，「諸言」，「序論」である．各学会誌の書き方が異なっているが，JSiSE の構造を特化した正規表現を使用すれば，領域分割が可能になる．具体的な領域分割方法については，表 2 に記載する．

表 2 学術論文の領域分割方法

パターン	分割方法
基本情報領域	**・本文`はじめに`前のテキスト**
本文領域	**・正規表現：章節番号 1 and 文字列パターン** ^(1)¥s*(は¥s*じ¥s*め¥s* に)¥s*$|(1)¥s*(ま¥s*え¥s*が¥s* き)¥s*$|^(1)¥s*(諸¥s* 言)¥s*$|^(1)¥s*(序¥s*論)¥s*$
参考文献領域	**・正規表現：文字列マッチ** ^(文¥s*献)¥s*$|^(参¥s*考¥s*文¥s* 献)¥s*$

3.2.3 文字列のフォントやサイズ特徴によるコンテンツ分割

節 3.2.2 に従って領域を分割した後，領域内のテキストコンテンツに対する階層分類を行う．そのため，テキストの付属情報を取得することが前提となる．

Pymupdf でテキストブロックを抽出すると同時に，フォント，サイズ，書体，カラーなどのテキスト付属

情報が得られる．次は，その付属情報を利用して，文章群，章節，図表などの論文内部コンテンツを分割するため，表 3 のようにコンテンツ分類器を構成する．

表 3 JSiSE 論文向けのコンテンツの認識方法

情報種類	コンテンツ種類	認識方法
本文情報	文章群	・フォント指定 KozMinPro-Light
補助情報	章節	・大項目，中項目：フォント指定 KozGoPro-Medium
		・小項目 ①フォント指定 KozMinPro-Light ②正規表現 ^[0-9]{1}(¥.[0-9]{1,2}){2,}¥s+.*$
補助情報	図	・図タイトル： ①正規表現 ^図¥s*¥d+¥s* ②フォント指定 KozMinPro-regular
		・図内容：図ブロックパターン ^<image:.*width:.*height:.*>$
補助情報	表	・表名： ①正規表現 ^表¥s*¥d+¥s* ②フォント指定 KozMinPro-regular
		・表内容：組み合わせて判定 ページの末尾には表領域が配置されている．ページに複数の表が存在する場合は，正規表現を組み合わせて，表領域を抽出することができる．

3.2.4 文章分割

機械学習タスクで使用される文章データを自動構築するため，本節では，節 3.2.3 の流れの通りに収集した本文テキストを対象とする文章分割処理を行い，文章群データの構築方法を紹介する．基本的な方法としては，句点パターン「.」をマッチして文章分割を行うが，句点が混在する複数のパターンがあるため，正規表現を使って個別に認識し処理する必要がある．これらのパターンは，メールアドレス，URL，複数の句点，項目リストである．

文章分割処理の流れを図 6 に示す．最初に特殊なパターンに対するフィルタリングを行い，特殊なパターンに対応する文字列にマスクを付け，記憶する．その後，単一の句点パターンで文章分割を行う．最後に，記憶した文字列を元の位置に復元して処理を終了する．

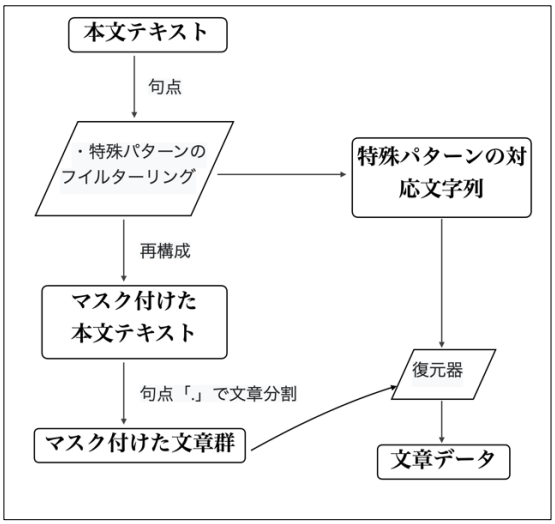


図 6 文章分割の流れ

3.3 データセットの構成

本節では，論文構造を反映したデータセットの構築について述べる．論文構造を反映するために，論文コンテンツ情報とメタ情報を組み合わせて，文章単位までの階層化を行う．また，コンテンツ間の所属関係を明示することで，最終的なデータセットを構築する．データセットの構成は図 7 に示す．

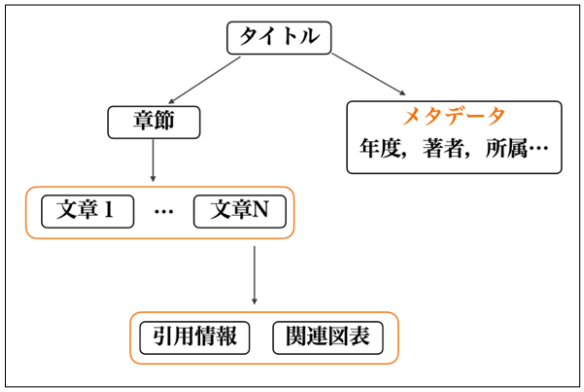


図 7 データセットの構成

4. 評価実験

4.1 実験方法

本研究では，スクレイピングで収集された 511 個の文献 PDF ファイルから 452 編の研究論文，実践論文，解説，ショートノートを実験対象とした．2012 年から 2021 年までの年度毎に 2 編の論文をサンプリングし，合計 20 編の PDF ファイルを実験用の文献ファイルとして設定した．テキスト前処理の有効性を検証するために，章，本文，図表の抽出に向けて，英文向けの科学論文から構造化メタデータを抽出するためのモジュール Cermine[17]，GROBID[18,19]，を比較対象として実験を行った．実験の評価基準は Horacio らがまとめた「科学出版物の構造分析」[20]を参考にして以下の 4 段階評価とした．

レベル 1：**利用可能なファイルである（PDF を入力できる）**

レベル 2：**テキストが再現できる**

レベル 3：**ある程度論文構造解析できる**

レベル 4：**ほぼノイズなしに論文構造解析できる**

上記の 4 段階レベルの設定に基づいて，手法ごとに各レベルの割合を算出する．

4.2 実験結果

サンプリングした 20 編の文献 PDF に対する比較実験の結果は表 4 に示す通りである．

表 4 比較実験の結果

手法	レベル評価	割合
提案手法	レベル 1：0 編	レベル 1：0%
	レベル 2：0 編	レベル 2：0%
	レベル 3：1 編	レベル 3：5.0%
	レベル 4：19 編	**レベル 4：95.0%**
Grobid	レベル 1：0 編	レベル 1：80.0%
	レベル 2：2 編	レベル 2：10.0%
	レベル 3：18 編	レベル 3：90.0%
	レベル 4：0 編	レベル 4：0%
Cermine	**レベル 1：16 編**	**レベル 1：80.0%**
	レベル 2：1 編	レベル 2：5.0%
	レベル 3：3 編	レベル 3：15.0%
	レベル 4：0 編	レベル 4：0%

上記の結果から見ると，本提案手法がレベル 4 に対して最も高い割合を示している．このことから JSiSE 論文誌を対象とした日本語論文に対する前処理の構造解析の有効性が確認された．その原因としては，Pymupdf ライブラリが日本語論文 PDF に対応していることや，日本語向けの正規表現が適切に使用されると考えられる．

4.3 考察

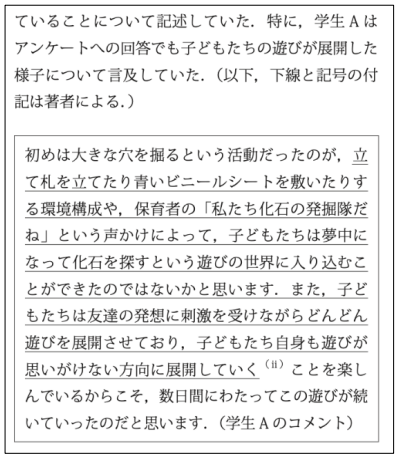
4.3.1 ウェブスクレイピングの問題点

511 個の収集した PDF ファイルのうち，59 個は学術論文の原稿ではなく，学会誌の表紙，通知，査読者情報などのファイルである．これらのファイルは評価の際に手作業で除外したが，今後はウェブスクレイピングの改善のポイントとして，論文原稿ではないファイルを自動的に判別する手法の検討が必要である．

4.3.2 テキスト前処理の問題点

(1) アンケート文の混在

図 8 に示すように，本文テキスト内には本文と異なるアンケートデータが混在している．そのような本文らしくないコンテンツを正確に処理できていないことがわかる．


ていることについて記述していた．特に，学生 A はアンケートへの回答でも子どもたちの遊びが展開した様子について言及していた．（以下，下線と記号の付記は著者による．）

初めは大きな穴を掘るという活動だったのが，立て札を立てたり青いビニールシートを敷いたりする環境構成や，保育者の「私たち化石の発掘隊だね」という声かけによって，子どもたちは夢中になって化石を探すという遊びの世界に入り込むことができたのではないかと思います．また，子どもたちは友達の発想に刺激を受けながらどんどん遊びを展開させており，子どもたち自身も遊びが思いがけない方向に展開していく(ii) ことを楽しんでいるからこそ，数日間にわたってこの遊びが続いていったのだと思います．（学生 A のコメント）

図 8 アンケート文の混在[21]

(2) 文章分割の際に不規則の箇条書き

節 3.2.4 で述べた句点が混在している特殊なパターンに対する正規表現でのノイズ認識や解除を行ったが，逆順の箇条書き（最初のパターンが「1.」ではない場合）などのレアなパターンには対応していない問題があった．

5. おわりに

本稿では，JALC データベースに掲載された学術論文を対象に，DOI の番号ルールに基づいて JSiSE 論文誌を選定し，論文毎のメタデータや文献ファイルを自動収集するためのウェブスクレイピングを行った．加えて，機械学習タスクで利用可能なデータセットを自動的に構築するため，データセット自動収集システムを開発した．これにより日本語文献ファイルの前処理の作業量を大幅に削減することができた．最後に前処理の効果を検証するための比較実験を行った結果，本手法は日本語文献ファイルに対する前処理の効果により，本システムの有効性が示された．

今後の課題は以下の通りである．

① ウェブスクレイピングの機能を拡張するために，新しい論文の更新に対する同期や通知機能を開発する．

② 本研究では，JSiSE 文献ファイルのコンテンツの自動分類のために，PDF 解析の段階では Pymupdf を使用したが，他の論文誌に対する前処理では，その論文誌の共通構造や書体に応じたテンプレートを構築することが必要である．そのため，汎用性の向上を目的とした統合テンプレートの設計・開発が今後の課題となる．

謝辞

本研究は JSPS 科研費 JP20H04295 の助成を受けた．

参考文献

(1) Li Jinghong, 太田光一, 長谷川忍:観点を反映した深層学習および強化学習による学術論文の自動要約生成, 教育システム情報学会研究報告, Vol36, No.1, pp.68-73, 2021.

(2) 中智宏, 漆原宏丞, 本多佑希, & 兼宗進. (2021). オープンデータを授業利用するためのスクレイピング WebAPI の開発. 情報処理学会第 83 回全国大会, 1, 07.

(3) 松延千春, 石井起弥, 井上寛, & 白谷智宣. Web スクレイピングを利用した医薬関連情報の収集と入手データの活用. 第一薬科大学研究年報, (38), 51-64.

(4) 一般向けデータ提供サービス利用規約 － ジャパンリンクセンター：https://japanlinkcenter.org/top/doc/JaLC_general_riyoukiyaku.pdf

(5) ジャパンリングセンター (JALC) との連携強化, JALC, DOI について：https://www.jstage.jst.go.jp/static/files/ja/JaLCrenkeikyouka.pdf

(6) 著作権法の一部を改正する法律　概要説明資料 https://www.bunka.go.jp/seisaku/chosakuken/hokaisei/h30_hokaisei/pdf/r1406693_02.pdf

(7) 上野達弘. (2021). アーティクル: 情報解析と著作権―「機械学習パラダイス」としての日本. 人工知能, 36(6), 745-749.

(8) Wahaj Salem Alkaberi, Reem Hamed Aljuhani, and Huda Mohamed Alamoudi. 2022. Web Scraper Application for Extracting Scientific Journals Data. In The 5th International Conference on Future Networks & Distributed Systems (ICFNDS 2021). Association for Computing Machinery, New York, NY, USA, 220–224. https://doi.org/10.1145/3508072.3508106

(9) 久保琢也, 伊藤広幸. J-STAGE を活用した日本の学術論文データの整備. 情報誌「大学評価と IR」第 12 号 令和 3 年(2021 年)9 月. [事例報告](大学評価コンソーシアム)

(10) C. Clark and S. Divvala, "PDFFigures 2.0: Mining figures from research papers," 2016 IEEE/ACM Joint Conference on Digital Libraries (JCDL), Newark, NJ, USA, 2016, pp. 143-152.

(11) Yang, H. (2022, November).PIEKM:ML-based Procedural Information Extraction and Knowledge Management System for Materials Science Literature. In Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing: System Demonstrations (pp. 57-62).

(12) 武田英明. (2021). 学術における永続的識別子としての DOI のこれまでと今後について. 情報の科学と技術, 71(4), 177-180.

(13) 三村のどか. (2022). JaLC メタデータのオープン化.

(14) REST API 情報提供機能の説明: https://japanlinkcenter.org/top/doc/REST_API_Functional_Description.pdf

(15) N. A. Sultan and D. B. Abdullah, "Scraping Google Scholar Data Using Cloud Computing Techniques," 2022 8th International Conference on Contemporary Information Technology and Mathematics (ICCITM), Mosul, Iraq, 2022, pp. 14-19, doi: 10.1109/ICCITM56309.2022.10032044.

(16) PyMuPDF Documentation：https://pymupdf.readthedocs.io/en/latest/toc.html

(17) Tkaczyk, D., Szostek, P., Fedoryszak, M. et al. CERMINE:

automatic extraction of structured metadata from scientific literature. IJDAR 18, 317–335 (2015). https://doi.org/10.1007/s10032-015-0249-8

(18) Lopez, P. (2009). GROBID: Combining Automatic Bibliographic Data Recognition and Term Extraction for Scholarship Publications. In: Agosti, M., Borbinha, J., Kapidakis, S., Papatheodorou, C., Tsakonas, G. (eds) Research and Advanced Technology for Digital Libraries. ECDL 2009. Lecture Notes in Computer Science, vol 5714. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-04346-8_62

(19) GROBID Documentation：https://grobid.readthedocs.io/en/latest/

(20) Horacio Saggion and Francesco Ronzano. 2016. Natural Language Processing for Intelligent Access to Scientific Information. In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Tutorial Abstracts, pages 9–13, Osaka, Japan. The COLING 2016 Organizing Committee.

(21) 藤原伸彦, 田村隆宏, & 木下光二. (2012). 「遊誘財データベース」 を活用した保育者養成. 教育システム情報学会誌, 29(1), 80-85.