国立情報学研究所教育研修事業
「大学図書館員のための IT 総合研修」2020 年度
「Web API を使ったデータの入手とその整備」講義資料

図書館 Web API のための OpenRefine 活用法(2)

～基本的な機能と Web API 取得データのクレンジング～

2020 年 9 月

東京大学情報システム部　前田朗

筑波大学附属図書館　松野渉

内容

1. 単純なデータ処理

Web API のデータをもとに、まずは Excel のようなスプレッドシートでもできる単純なデータ処理から説明する。

1.1 Web API によるデータの取得

実習 1.1-1： 国立国会図書館サーチ Web API からデータを取得する

① Web ブラウザから以下の URL でデータを取得する
https://iss.ndl.go.jp/api/opensearch?any=吾輩は猫である&cnt=200

② OpenRefine で取得したファイルを取り込む

③ OpenRefine のデータ取得範囲を**<item>**にし、プロジェクトを作成する

> **補足： レコード単位が正しく判定できないデータもある**
>
> ・試した限りでは、国立国会図書館サーチと CiNii Books (Atom)の OpenSearch 取得データがある。
>
> ・JAIRO Cloud の OAI-PMH の取り込みを試したときは、プロジェクト作成後の「**カラムの並び替え**」により、「レコード」単位が正しく判定されるようになった。

実習 1.1-2： カラムの並び替えによりデータをみやすくする

> **ポイント： カラム名の「-」は階層分け**
>
> ・OpenRefine は元データの項目名をもとにカラム名を自動付与する。階層構造の場合は、「-」で区切って生成する。
>
> ・カラムの並び替えにおいては、この「-」をもとに階層ごとに整理することを薦めたい。

1.2 値の加工

実習 1.2： セルの値を変更する

① 値のあるセルにカーソルを合わせ、セル右上に表示された「**edit**」をクリックすると、セルの編集欄が表示される

② セルの編集欄の「**データの型**」を確認する

③ セルの値を修正する

④ 「**適用**」と「**同じ内容のセルに適用**」のボタンのうち「適用」を選び、値を修正する

補足： OpenRefine のデータ型

・OpenRefine には以下のデータ型がある。この資料では取り上げないが、どの「**数値ファセット**」などテキスト以外の処理をかけたいときは変換をする必要がある。

- テキスト
- 数値
- 論理値
- 日付

1.3 テキストフィルタ

実習 1.3： 出版者に「**日本**」を含む行のみ表示させる

① ヘッダ行の_item-dc:publisher カラムの▼をクリックする

② プルダウンメニューから「**テキストフィルタ**」を選ぶ

③ テキストフィルタに日本を入れ、その条件で表示レコードが絞られることを確認する

※テキストフィルタの右上「反転」を選ぶとその条件に**あてはまらないものを絞り込む**ことが出来る。今回の例であれば**「日本」を含まない行のみ**を表示できる

補足： 正規表現

・OpenRefine は「**テキストフィルタ**」や「**置換**」などで、正規表見による文字列のパターンマッチ指定ができる。

・正規行健の例は次のとおり。

- 「日本」から文字列が始まる（先頭一致） ⇒ ^日本
- 「大学」で文字列が終わる（末尾一致） ⇒ 大学$
- 「国立 （中略）研究所」を探す（中間任意） ⇒ 国立.+研究所

・例えば、ISBN のパターンマッチについては以下に情報がある。

https://github.com/OpenRefine/OpenRefine/wiki/Recipes

1.4　置換

実習 1.4：　出版者名に含まれる「日本」を「Nihon」に一括変換する

① ヘッダ行の_item-dc:publisher カラムの▼をクリックする
② プルダウンメニューから「**セル編集**」⇒「**置換**」を選ぶ
③ 置換元を「**日本**」に、置換後を「Nihon」に指定し、「OK」ボタンをクリックする
④ 値が一括で書き換わったことを確認する

1.5　ソート

実習 1.5：　タイトル名で行をソートする

① ヘッダ行の_-title カラムの▼をクリックする
② プルダウンメニューから「**ソート**」を選択する
③ ソート結果の先頭数行を確認する

> **ポイント：　ソートの効果**
> ・文字列の先頭が記号のものを確認できる。
> ・通覧することで名寄せ候補の文字列を確認できる。

1.6　GREL(General Refine Expression Language)の活用

GREL は OpenRefine 内でデータを処理するための言語である。Excel における関数に相当する。詳細は以下のサイトを確認のこと。

https://github.com/OpenRefine/OpenRefine/wiki/GREL-Functions

実習 1.6：　タイトル名の文字数を新規カラムで確認する

① ヘッダ行の_-title カラムの▼をクリックする
② プルダウンメニューの「**カラムを編集**」⇒「**このカラムに基づいてカラムを追加**」を選択
③ 新規カラム名を「**タイトル文字列数**」をする
④ 変数 value に対し、文字数をカウントする GREL コードを設定する
length(value)

補足：　GREL の利用について

・Excel の関数のように使える。

・if 文を使って簡単なロジックを組むことができる。

・GREL コードを入力する画面で「ヘルプ」を選択することで、利用可能な関数を確認できる。

補足：　GREL による文字列操作の例

・Excel の関数のように使える。

・文字列「A」を文字列「B」に置換。

```
value.replace(‘A’,‘B’)
```

・セルの値を文字列「A」で配列に分割。

```
value.split(‘A’)
```

2　ファセットとクラスタリング

OpenRefine ならではのファセットとクラスタリングの両機能について説明する。

2.1　ファセット

情報検索システムにおけるファセット検索のファセットと同義である。OpenRefine のデータクレンジングにおいては特定のデータ項目をグルーピングして集計するものと理解すればよい。

実習 2.1：　出版者名のファセットを作成する

① ヘッダ行の_item-dc:publisher カラムの▼をクリックする
② プルダウンメニューから「**ファセット**」⇒「**テキストファセット**」を選ぶ
③ 左側メニューに表示されるファセットを確認する
④ ファセットを選択することで表示される行を絞り込めることを確認する

ポイント：　ファセット機能の利用

・コードや統制語彙の誤記チェックや名寄せに有効。
・テキストファセット →　ファセットが多い場合「選択肢が多すぎて表示できません」とのメッセージがでるが、「カウントを制限してください」との入力欄も表示され、そこに数値をセットすることとで上限を増やせる。OpenRefineの「**設定**」にある**ui.browsing.listFacet.limit**　で設定が有効なことが確認できる。
・複数のファセットを用いて絞り込みを行いたいときはファセット上にカーソルを移動させると出現する「**include**」という項目を選択する。
・逆にファセットの選択を解除したい際は「**exclude**」を選択する。

補足：　カスタムファセットあれこれ

・**重複ファセット**　→　ユニークキーの重複確認に使える。
・**単語ファセット**　→　単語判定が半角スペース区切り前提のため注意。
・**Unicode ファセット**　→　文字によるファセットのため時間がかかる。多くの文字からなる日本語向きではないかもしれない。

2.2　クラスタリングによる名寄せ

ファセット化したデータ項目をさらにクラスタリング（類似のものをまとめる）ことで、名寄せを行うことができる。クラスタリングにはいくつものパラメータがあり選択・調整して活用する。

実習 2.2：　出版者名ファセットをクラスタリングする

① 出版者ファセット欄の「**クラスタ**」をクリックする
② クラスタが表示されない場合、パラメータを変更する
③ 類似の文字列が検出されることを確認する
④ 「マージ」ボタンにより名寄せができることを確認する

ポイント：　クラスタリングのパラメータ設定の調整

・「**キー衝突法**」
- 著者の姓名の順序チェックに使えた。
- 「キー衝突法」の **Fingerprint** は「指紋」のこと。文字列から特徴を抽出（「文を単語区切り」⇒「大文字・小文字の統一」⇒「単語重複削除」⇒「単語を文字コード順に文字列連結」）し照合。　→　動作が早いことがメリット。
- 「キー衝突法」は単語半角スペース区切りが前提なので、**日本語の場合は**キー関数を n-gram にするか、「最近傍法」がよいかもしれない。

・「**最近傍法**」
- 「最近傍法」の「**レーベンスタイン距離**」は文字列を何回編集すると一致するかで類似度を出す。よりあいまいにマッチさせたい場合は、「**ngram 半径**」を大きく、「**文字ブロック**」を小さく設定すること。
- サブタイトルの有無をチェックするには「最近傍法」かつ「**ppm**」にするのが有効に思えた。

・OpenRefine のクラスタリングについてより詳しくは以下を参照のこと。
https://github.com/OpenRefine/OpenRefine/wiki/Clustering-In-Depth

ポイント：　クラスタリングに向いてないと思われること

・異体字を含む文字列の名寄せ。
・巻号付きのタイトル。　→　最近傍法ではノイズが多く出る。
・Excel のフィル機能などでコピーと間違えて連番にしたデータの検知。

3　一括処理

OpenRefine における一括処理の方法を提示する。

3.1　ファセット指定による一括修正

実習 3.1：　特定の出版社名を一括変換する

① 直前(実習 2.2)に作成した出版者名のファセットから、一括編集したい出版社名にカーソルを合わせる

② 「edit」が表示されたらクリックし、変更する値を入れて、「OK」をクリックする

3.2　スターとフラグ（旗）によるチェックと一括修正

実習 3.2-1：　レコードにスターとフラグ**を付ける**

③ よいレコードにスターを付ける

④ 問題レコードにフラグを付ける

実習 3.2-2：　フラグを付けたレコードを一括削除する

① カラム「**全て**」の▼メニューから、「**ファセット**」⇒「**旗ファセット**」を選択する

② 「**旗ファセット**」の ture の値のみ表示させる

③ カラム「**全て**」の▼メニューから、「**行を編集**」⇒「**マッチした行を削除**」を選択する

④ フラグをつけたレコードが削除されたことを確認する

補足：　スターとフラグの違い

・システム上は差がなく、よいデータにフラグ、悪いデータにスターをつけても使用はできるようである。

4　外部データとの照合

OpenRefine ならではの機能として強力な外部データ照合の機能がある。

4.1　照合の基本

外部データを使った照合ができる。照合結果は照合を行ったカラムに追加情報として表示される。照合したカラムをもとに、以下の GREL 式を指定することで別カラムも表示できる

・第一候補の id

```
cell.recon.candidates[0].id
```

・第一候補の名称

```
cell.recon.candidates[0].name
```

・閾値 0.7 以上のみ第一候補名称を表示(RDF Refine と Reconcile-csv の場合のみ)

```
if (cell.recon.candidates[0].score >= 0.7, cell.recon.candidates[0].name
<"" )
```

4.2　OpenRefine の照合機能

デフォルトは Wikidata(データ版の Wikipedia)の英語版のみであるが、以下のサービスなどの設定を追加することで利用できるようになる。

【Wikidata 日本語版】

```
https://wdreconcile.toolforge.org/ja/api
```

【VIAF（バーチャル国際典拠ファイル)】

```
http://refine.codefork.com/reconcile/viaf
```

4.3　RDF との照合

RDF Refine 拡張を使えば可能。ローカルの RDF ファイル読み込みと、SPARQL エンドポイント指定のいずれもサポートしている。ローカルの RDF ファイル読み込みには時間がかかる。

4.4　CSV との照合

Reconcile-csv を使う方法がある

http://okfnlabs.org/reconcile-csv/

5　プロジェクトの保存

OpenRefine では処理が自動で保存されるため、アプリケーションを終了する際に保存をする必要はない。OpenRefine の次回起動時に左側パネルの「**プロジェクトを開く**」から以前に作成したプロジェクトを呼び出せる。

5.1　プロジェクト全体の保存

OpenRefine 画面右上の「**出力**」から「**プロジェクトのエクスポート**」を選択。出力したプロジェクトデータは別の PC 環境に取り込んで使う、バックアップにするといった使い方が考えられる。

5.2　スプレッドシートとして保存

「図書館 Web API のための OpenRefine 活用方(1)」の「スプレッドシートで出力」を参照。

5.3　作業履歴の保存

OpenRefine 画面左パネルの「**取り消す/やりなおす**」⇒「**抜き出し**」を実行。保存した作業履歴は、「**抜き出し**」の右隣にある「**適用**」から別プロジェクトに適用するといった使い方が考えられる。

6　文献データのデータクレンジングの着目点

別紙「東京大学の機関リポジトリデータクレンジング事例」参照。OpenRifine を使ってはいないが、文献データに対し、どのような観点でデータクレンジングを行ったか、参考にしてほしい

7　OpenRefine によるデータクレンジングサンプル事例

詳細を知りたい場合は、講師に声をかけること。

7.1　機関リポジトリのデータを OpenRefine でクレンジング

・東京大学学術機関リポジトリの OAI-PMH データ 5 万 4 千件を実際に取り込み

・NDC 分類のファセット化、**junii2** の **jtitle** のクラスタ化による名寄せ、DOI 形式のチェックなど

7.2　ジャパンサーチ簡易 API で「葛飾北斎」の名寄せを試す

・デジタルアーカイブのデータはクレンジングのしがいがある

・ジャパンサーチの簡易 API では、全件取得のための機能(scroll)がある(SPARQL で

も検索結果全件取得はできる）

https://jpsearch.go.jp/static/developer/webapi/

・OpenRefine は PC 上のファイル・URL とも複数指定で取り込みが可

7.3　ERDB-JP の誌名と ISSN 日本センターの一覧を照合する

・ERDB-JP のデータクレンジング

→　ISSN 一覧とのチェックではなくレコードの重複のチェックのほうがクレンジングのしがいがあるかもしれない

・ISSN 日本センターの一覧データの活用

https://www.ndl.go.jp/jp/data/issn/index.html

・OpenRefine における CSV データとの照合

http://okfnlabs.org/reconcile-csv/

7.4　自機関機関リポジトリの subject から NDC8 分類を出す

・講習で使う openrefine-3.4 に、RDF Refine 3.3 拡張を適用できる

・RDF Refine 拡張を入れると、RDF と OpenRefine のカラムとの照合ができる

・RDF の照合は、ローカルの RDF ファイルを取り込んでも、SPARQL エンドポイントを指定でもよい

・NDC はファイルで RDF が公開されており、それを使用した

8　トラブルシューティング

8.1　OpenRfine 環境を初期化するには

・作業ディレクトリを削除し、OpenRefine を再起動すればよい

・作業ディレクトリは、「プロジェクトをつくる」→「作業ディレクトリを閲覧」で確認できる。Windows の場合、Windows のユーザーフォルダ（C:¥Users¥xxx）以下の次のフォルダ

¥AppData¥Roaming¥OpenRefine

8.2　ランサムウェア警告

・OpenRefine の操作中に、ウイルスバスターが PC 内のファイルを暗号化しているとして、ランサムウェア警告を出すことがある。

9　余談

9.1　OpenRefine のダイヤモンドアイコン

由来は以下と思われる。

“We prefer to think data as diamonds”

書籍 Using OpenRefine （ISBN: 9781783289080）より

9.2　OpenRefine “with embedded Java” の Java 環境

OpenRefine 3.4 Windows kit with embedded Java に含まれる Java 環境は、基本的に OpenRefine 以外で使うことができない。とはいえ、この Java 環境は自動で update されず、放置するとじきに古いバージョンとなるので、注意が必要であろう。他の Java アプリケーションでこの Java 環境を使うのは**あくまで裏技**となるが、Windows の場合、コマンドプロンプトで次のコマンドを実行（環境変数をセット）すれば動作した。

JAVA_HOME = *OpenRefine のディレクトリパス*¥server¥target¥jre
set PATH=%PATH%; *OpenRefine のディレクトリパス*¥server¥target¥jre¥bin

【参考情報】

Web サイト

1. OpenRefine
   https://openrefine.org/
2. OpenRefine (GitHub wiki)
   https://github.com/OpenRefine/OpenRefine/wiki
3. OpenRefine で OSM データを扱ってみた
   https://qiita.com/higa4/items/184c51de632fade29d6b
4. OpenRefine で神エクセルと戦う
   https://qiita.com/higa4/items/5c2b2630bfd91e064f67
5. OpenRefine と Word で PDF と戦う
   https://qiita.com/higa4/items/7cb4c833e383b47b2c18
6. OpenRefine と Word で PDF と戦う 2-名寄せ・照合編
   https://qiita.com/higa4/items/41abba38d5b33227e4a4
7. Open Refine for Librarians
   https://librarian.aedileworks.com/2019/04/23/open-refine-for-librarians/
8. Cleaning Data with OpenRefine

https://programminghistorian.org/en/lessons/cleaning-data-with-openrefine

9. OpenRefine を使ってデータをきれいにする方法
   https://www.dh.ku-orcas.kansai-u.ac.jp/?p=468
   (8 の和訳版)
10. Reconcilable Data Sources
    https://github.com/OpenRefine/OpenRefine/wiki/Reconcilable-Data-Sources
11. Library Carpentry: OpenRefine
    https://librarycarpentry.org/lc-open-refine/
12. RDF Refine の使い方
    https://www.slideshare.net/takeda/rdf-refine

動画サイト

1. Open Refine (旧 Google Refine) の使い方 ～導入・基本編 ...
   https://togotv.dbcls.jp/20130212.html
2. Open Refine (旧 Google Refine) の使い方 応用編・TogoWS REST を活用する
   https://togotv.dbcls.jp/20130530.html
3. Google Refine 2.0 - Introduction (1 of 3) (video version 2)
   https://www.youtube.com/watch?v=B70J_H_zAWM
4. Google Refine 2.0 - Data Transformation (2 of 3) (video version 2)
   https://www.youtube.com/watch?v=c08NVCs_Ba0
5. Google Refine 2.0 - Data Augmentation (3 of 3) (video version 2)
   https://www.youtube.com/watch?v=5tsyz3ibYzk

書籍

1. Using OpenRefine (ISBN: 9781783289080)
   https://www.packtpub.com/big-data-and-business-intelligence/using-openrefine