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(1)

～インストールから Web API 取得データの Excel 化まで～

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

目次

1　はじめに

1.1　OpenRefine とは

OpenRefine は、取り込んだデータに対し、一括修正・名寄せ・外部データとの照合といったデータクレンジングを行えるアプリケーションである。

操作画面上は Excel のようなスプレッドシートにも見えるが、実態はスプレッドシートとデータベースのハイブリッドである。スプレッドシートとは異なり、データの直入力には向いていない。クレンジング対象のデータを外部から取り込み使うべきである。また、繰り返しのデータ項目を持つレコードを１レコード複数行として、取り扱うことができる。

OpenRefine でクレンジングしたデータを、そのままデータの取得先に一括反映できるのが望ましいが、修正すべきデータのチェックだけでも役立つはずである。

1.2　この資料について

OpenRefine のバージョンは、**OpenRefine 3.4 Windows kit with embedded Java** を使用しており、説明もそれに基づく。Windows10 環境での操作説明となるが、Machintosh でも同様の操作が可能なはずである。説明の内容は講師が実際に試したことと、参考文献から得た情報を基にしている。

2　OpenRefine のインストール

2.1　プログラムのダウンロード

実習 2.1：　OpenRefine のサイトか Java 環境込みのバージョンをダウンロード

① OpenRefine のダウンロードページにアクセスする
https://openrefine.org/download.html

② 以下のバージョンを選択してダウンロードする
OpenRefine 3.4 Windows kit with embedded Java

③ ダウンロードした zip ファイルを解凍する

> **補足：　OpenRefine の Java 環境**
> ・OpenRefine は Java 実行環境で動作する。
> ・**with embedded Java** のバージョンは、Java 実行環境が梱包されており、別途インストールする必要はない。

2.2　サーバプログラムの起動

実習 2.2：　OpenRefine を起動する

① 直前の実習(実習 2.1)で作成された「**openrefine-3.4**」フォルダにある”**openrefine.exe**”（青いダイヤモンドのアイコン）をクリックし、プログラムを起動する

② Web ブラウザが起動し、OpenRefine の初期画面が出ることを確認する

> **ポイント：　ローカルで動く Web アプリケーション**
> ・OpenRefine はローカルで動く Web アプリケーションになっており、これがサーバプログラムである。このウインドウを閉じると OpenRefine も停止する。
> ・Web API を備えるため、PC 上のプログラム（Python 等）から操作することもできる。

補足：　インストール時の警告メッセージ

・最初の起動時の「発行元が不明」の警告は無視する(OpenRefine はコード証明をとっていないらしい)。

・ローカルでサーバを立ち上げる際に警告がでる可能性がある。

補足：　OpenRefine が利用できるメモリを増やす

・OpenRefine の起動設定ファイル”**openrefine.l4j.ini**”を「メモ帳」などのテキストエディタで編集することで、Java の使用可能なメモリサイズ(デフォルトは 1GB) を増やせる。

2.3　アプリケーションの表示言語設定

実習 2.3：　OpenRefine の表示言語設定をする

① 左側メニューの”**Language Settings**”をクリックし、アプリケーションの表示言語を初期表示言語の”**English**”から”**日本語**”に変更する

補足：　OpenRefine の動作設定

・直に操作する必要性は薄いはずであるが、トップ画面の「設定」からファセットの最大数など動作設定が可能となっている。

3　Web API の取得データの取り込み

3.1　Web API データ取得結果ファイルからの取り込み

実習 3.1-1： CiNii Articles Web API からデータを取得する

① Web ブラウザから CiNii Articles Web API に次のリクエストを行う
https://ci.nii.ac.jp/opensearch/search?q=吾輩は猫である&count=200&start=1&format=json&appid=*xxxx*
※「xxxx」には 1-1 で CiNii Web API の利用申請した際に取得したアプリケーション ID を挿入する

② データ取得結果をファイルに保存する。多くのブラウザでは画面上でマウス右ボタンクリックにより保存メニューがでる

実習 3.1-2： OpenRefine にファイルからデータを取り込む

① 中央パネルの「ファイルの選択」から、直前（実習 3-1.1）で取得したファイルを取り込む

② Web API の取得データが表示されることを確認する

補足： OpenRefine の複数ファイル取り込み
・OpenRefine では「ファイルの選択」の際に、フォルダ内のファイルをまとめて指定できる。
・Web API 取得データが複数ファイルにわかれているときに有用。

補足： UTF-8 の CSV ファイルは事前に BOM を削除
・UTF-8（BOM あり）の CSV ファイルは、テキストエディタなどで事前に「BOM なし」にしないと、正しく OpenRefine に取り込めない。
・Web API の講習で取り上げた「6 Web ブラウザで使える文献 Web API 取得結果のスプレッドシート化」のデータを取り込むときは、この BOM の削除が必要となる。

補足： Excel ファイルを正しく取り込めないときは
・TSV ファイルなど別形式に変換して取りこむ対応がある 。

3.2　インターネットから URL で直接取り込み

実習 3.1-3：　OpenRefine にインターネットから URL 指定で直接データを取り込む

① 「**最初からやりなおす**」⇒「**ウェブアドレス（URLs)**」で、URL 入力欄を表示する

② Web 画面の URL 入力欄から実習 3.1-1 の Web API リクエスト URL をコピーする

③ URL を①の URL 入力欄に張り付け、「**次へ**」ボタンをクリックする

④ ファイルからの取り込みと同じく、Web API 取得結果が表示されることを確認する

> **補足：　OpenRefine の複数 URL 指定取り込み**
> ・OpenRefine では URL 入力欄で「**次へ**」をクリックすると、次の URL 入力欄が表示され、複数 URL をまとめて指定できる。

3.3　データ解析形式の指定

実習 3.3：　OpenRefine でデータ解析形式を指定する

① データ解析形式(**XMLFiles, JSON Files** など)が、正しく自動判別されたか確認する

② 正しく判別されない場合は手動で設定する

> **補足：　SRU の取り込みには事前データ加工を**
> ・国立国会図書館サーチで試した限りでは SRU のデータの取り込みには事前加工が必要であった。
> ・OpenRefine でレコードデータのみを取り出し　⇒　PC にエクスポート　⇒　XML 形式に変換　⇒　OpenRefine に取り込み、により対処はできた。

3.4　レコード範囲の指定とプロジェクトの作成

実習 3.4：CiNii Books Web API (JSON 形式)のレコード範囲を指定する

① OpenRefine の CiNii Books Web API (JSON 形式)のレコード範囲の最初のレコードを指定する。指定範囲は色が変わる。

② 指定範囲に問題がなければクリックする

③ プロジェクトのプレビュー画面で、データ項目をカラムとした表形式でデータが解析されたかを確認する

④ 問題がなければ「**プロジェクトの作成**」へ、問題があれば「**レコードのパスを指定してください**」ボタンで指定をやりなおす

> **補足：　特定のデータ項目のみの一覧を作れる**
> ・レコードの範囲指定ではなく、たとえば「タイトル」など特定の項目を指定することで、特定の項目のみの一覧を作れる。

3.5　プロジェクトの操作画面の確認

実習 3.5：プロジェクトの操作画面を確認する

① ヘッダ行にデータ項目名が入っていることを確認する

② ヘッダ行のカラム名称は先頭しか表示されないが、カーソルを合わせることですべて表示されることを確認する

③ ヘッダ列にある▼をクリックすることによりカラムの操作メニューが表示されたことを確認する

④ 一番左のカラムが「**全て**」のセルの操作用のカラムであることを確認する

4　スプレッドシートで出力

4.1　「行」と「レコード」

実習 4-1：　「行」と「レコード」の違いを確認する

① 表示の「**レコード**」を「**行**」に切り替える
② 1 レコード複数行になっていることを確認する
③ 表示を「**50**」行にして一覧を閲覧することで、どのカラムが繰り返し項目になっているのかを確認する

> **補足：　「レコード」が正しく認識されないときは**
> ・1 レコード 1 行にしてスプレッドシートへの出力は難しいが、問題データのチェックには支障がない。
> ・後述の「**カラムの並び替え**」により「レコード」が正しく認識されることがある。

4.2　カラムを選定し並び順を変更する

実習 4-2：　カラムの選定と並び順の変更法を行う

① 「**全て**」⇒「**カラムを編集**」⇒「**カラムの並び替え・削除**」にて、「カラムの並び替え・削除」画面を表示させる
② わかりやすいようにタイトルなどの情報が左側にくるようにする
③ 不要なカラムを指定してもよい
④ 「**OK**」をクリックし、表示カラムと並び順が変更されたことを確認する

4.3　スプレッドシート出力の前処理で 1 レコード 1 行にする

実習 4-3：　1 レコードを 1 行に変換する

① 繰り返し項目になっているカラムを確認する
② ヘッダ行から該当のカラムの▼をクリックする
③ プルダウンメニューから「**セルの編集**」⇒「**多値のセルを結合**」を選ぶ
④ 区切り文字を”,”から”|”に変更し「**OK**」をクリックする
⑤ 繰り返し項目が一つのセルに”|”区切りで入ったことを確認する
⑥ 表示を「**行**」と「**レコード**」で切り替え、件数を確認する

4.4　データクレンジングに向く 1 レコード複数行にする

実習 4-4：　区切り記号により多値を入れているセルを分割する

① 直前（実習 4-2）で処理しカラムについて、ヘッダ行から該当のカラムの▼をクリックする
② プルダウンメニューから「**セルの編集**」⇒「**多値のセルを分割**」を選ぶ
③ 区切り文字を”|”に変更し「OK」をクリックする
④ 区切り記号で複数値を入れているセルが複数行に分割されたことを確認する
⑤ 表示を「**行**」と「**レコード**」で切り替え、件数を確認する

4.5　作業履歴をもとに処理を元に戻す

実習 4.5：　1 レコード 1 行にした状態まで戻す

① 左側パネルの「**取り消す/やり直す**」をクリックする
② いままでの操作履歴が表示されることを確認する
③ 1 レコード 1 行のデータにした時点まで戻す

4.6　スプレッドシート形式で出力

実習 4.6：　OpenRefine のプロジェクトを Excel 形式で出力する

① 右上の「出力」ボタンをクリックする
② プルダウンメニューで出力形式を Excel に指定すると、出力結果がダウンロードされる
③ 出力結果を Excel で確認する

補足：　スプレッドシートへの出力後のマージは避ける

・Web API は値があるデータ項目名しか出力しないことが多い。そのため、複数回に分けて Web API で取得したデータ項目が、毎回同じかどうかの保証はしきれない（たまたま出現した項目しか処理されない）。
・これを避けるために、OpenRefine の複数ファイルの取り込みにより、まとめて処理することを薦めたい。