図書館員向けに考えたPython自習課題

Lesson 1

「図書系職員のためのアプリケーション開発講習会」資料
2018年7月23日　東京大学情報システム部　前田　朗

よく使うデータ処理パターン

- TSVのテキストを書き換える
- TSVのテキスト同士をマッチングさせる

図書館業務でよく使うデータ処理のパターンです。
Pythonでどう解決するかをみていきます。

課題とコードサンプル

- 図書館の実データを使った課題をクリアしていきます
- 課題には、その課題の参考となるPythonコード例を示してあります。それを参考に、自身でプログラムをしてみましょう

前準備

- Python
  - Version 3系を使おう
  - せっかくなら、anacondaを。機械学習系のモジュールが最初から用意されている
- TSV
  - Utf-8にしておくと、そのままPython3で処理できる
  - Shift-JISになっている場合、テキストエディタなどで、utf-8で保存しておこう

課題1: TSVのテキストを書き換える

- 図書館システムから、自館の書誌・所蔵をTSVファイルで抽出する
- Pthonで次のとおりデータを抽出・加工する
  - 「NCID」・「タイトル情報」・「ISBN」の３つのカラムからなるTSVファイルを出力する
  - ISBNは-を取り除く

東京大学の図書館システム(e-cats)の場合、汎用ファイル出力機能を使えば、簡単にデータを用意できます。

基本形をもとに作ってみよう

```
# -*- coding: utf-8 -*-
input_file = "input.txt"      # 入力ファイルを指定
output_file = "output.txt" #    出力ファイルを指定
f_in = open(input_file, "r", encoding="utf-8")
f_out = open(output_file, "w", encoding="utf-8")

def myfilter(line):   # 自作の関数（先に定義しておくのがポイント)
        values = line.split("¥t")  # タブで値を分割してリスト values に入れる
        f_out.write(values[1])  # 2カラムめだけ出力
        f_out.write("¥n") # writeでは改行を明示的に指定する

for line in f_in:   # 入力ファイルから1行づつ読み込み
        line = line.replace('¥n','')  # windows改行コード削除1
        line = line.replace('¥r','')  # windows改行コード削除2
        myfilter(line)  # 自作の関数でデータ加工
f_in.close
f_out.close
```

正規表現を用いた置換

```
import re    # 正規表現のモジュールを読みこむ

issn = "1234-5678"
issn_new = re.sub(“-”, “”, issn)   # -を削除

print(issn_new)    # print文で結果を確かめる
```

課題2: TSVのテキスト同士をマッチングさせる

- データの用意
  - 図書館システムから、自館の書誌・所蔵をTSVファイルで抽出する（[リストA])
  - 図書館システムから、学内他館の所蔵資料の「NCID」と「巻次」の組みをTSVファイルで抽出する（[リストB])
- Pthonで[リストA]をベースに、リストBとマッチする書誌をチェックする

  → 学内他館でもっている自館料を確認できる（巻次レベルまで厳密は面倒ですが....）

東京大学の図書館システム(e-cats)の場合、汎用ファイル出力機能を使えば、簡単にデータを用意できます。

ディクショナリと条件分岐

- [リストB]のデータを読み込み、NCIDをキーにしたディクショナリをつくります
- 次に[リストA]を読み込み、NCIDについて、先に作成したディクショナリとマッチさせます。マッチの判定にはif文を使いましょう
- マッチした場合は、マッチしたことがわかるように[リストA]のレコード出力に加えてなんらかの値を出力させるようにします

ファイルからディクショナリをつくる

```
dic_file = "ncid.txt"     # NCIDのみのテキストファイル
f_dic = open(dic_file, "r", encoding="utf-8")

dict = {}     # ディクショナリを宣言しておく
for line in f_dic:
    line = line.replace('¥n','')
    line = line.replace('¥r','')
    values = line.split("¥t")   # タブで値を分割してリスト values に入れる
    ncid  = values[0]   # 1カラムめがNCIDとする
    dict[line] = line    # NCIDをキーにしたディクショナリ

f_dic.close
print(dict)    # 結果確認用のコード
```

次回予定

- TSVのテキストを別フォーマットに変換する
- CiNII Books APIからデータを取り出す