**情報工学実験IIレポート**

自然言語処理

人間情報システム工学科4年　45番

山口惺司

実験日： 2024年12月11日

2024年12月4日

締切： 2024年12月18日

提出日： 2024年12月18日

|  |  |
| --- | --- |
| 評価項目 | やった/一部やった/やっていない/何をやったかの概要 |
| **形態素解析の対象とした文書** | 斜陽 / 太宰治 |
| **正規表現の説明を行った** | 何を加えたか：英数字, 記号, 空白, 改行, 句読点, 読点, 括弧, の削除 |
| 追加的課題： | やってない |
| 追加的課題： | やってない |

**背景と実験目的**

背景：

自然言語処理が含む内容は極めて多岐にわたるが、今回の学生実験のテーマでは、Gensim というライブラリ群に入っている Word2Vec という比較的有名なツールを用いて、自然言語処理の一部を体験してみる。形態素解析、正規表現、分散表現(単語のベクトル化)などのいくつかの技術や概念に触れ、基礎的な知識を得る。

目的：

・正規表現の基礎的な機能を使うことができる

・ストップワードの基礎的な機能を説明できる

・形態素解析の基礎的な機能を使うことができる

**課題1**

【問題】

* 何らかのテキストに対して、形態素解析ツールを用い、分かち書きし、ファイルに出力する
* プログラム中の正規表現を用いて部分について、どのような処理を行っているか、説明を数行書く

**【アルゴリズム・解き方】**

今回は太宰治の「斜陽」を用いて、データの解析を行う。

1. 解析するテキストデータを用意する。
2. 正規表現を使って単語以外の文字をテキストデータから削除する。
3. Janomeを用いて分かち書きをし、ファイルに出力する。

正規表現では以下のような文字をテキストデータから削除している。

「英数字, 記号, 空白の行, 空白, 括弧, 句読点, 読点」

**【実行結果】**

実行結果は出力されたファイルの冒頭10単語の句読点なしを図1, 句読点ありを図2に示す.



図1 出力されたファイルの冒頭10単語



図2 出力されたファイルの冒頭10単語(句読点あり)

**【考察】**

　図1を見ると前処理で「、」をデータから削除しているため本来「朝、食堂でスウプを一さじ、….」という文が「朝食」と「堂」で分かれてしまい、意味が変わってしまっていることがわかる.

そこで, データから句読点を削除しなかった場合の実行結果である図2を見ると正しい単語の塊で出力されていることがわかる.

**まとめ**

　前処理の仕方次第で単語の処理が変わってしまうため、工夫が必要だということがわかった.

**参考文献**

　青空文庫：太宰治 斜陽  
https://www.aozora.gr.jp/cards/000035/files/1565\_8559.html

付録

**【プログラムソース】**

from janome.tokenizer import Tokenizer

import re

def preprocessing(text):

   text = re.sub("[a-xA-Z0-9\_]", "", text)

   text = re.sub("[!-/:-@[-`{-~\*]", "", text)

   text = re.sub(u'\n\n', '\n', text)

   text = re.sub(u'\r', '', text)

   text = re.sub(r'《.\*?》', '', text)

   text = re.sub(r'［.\*?］', '', text)

   text = re.sub(r"[「」？！。　]", "", text)

   return text

def save\_wakati\_file(wakati\_list, save\_path='wakati.txt'):

   with open('./' + save\_path, mode="w", encoding='utf-8') as f2:

       f2.write(' '.join(wakati\_list))

def main():

    tokenizer = Tokenizer()

    f1 = open('shayo.txt', 'r', encoding='UTF-8')

    s = f1.read()

    f1.close

    text = preprocessing(s)

    tokens = tokenizer.tokenize(text, wakati=True)

    w = list(tokens)

    save\_wakati\_file(w)

    print(w[1 : 11])

if \_\_name\_\_ == "\_\_main\_\_":

    main()