TF-IDF法を用いた国会議事録の分析

**Analysis of the minutes of the Diet using the TF-IDF method**

高橋　湧汰

Yuta Takahashi

法政大学情報科学部コンピューター科学科

*E-mail: yuta.takahashi.5t@stu.hosei.ac.jp*

# **1. 使用した手法**

Pythonのurlib.parseを用いて国会議事録APIから2020年1月1日～6月1日までの予算委員会において、安倍晋三首相の発言をスクレイピングし、それを元に形態素解析をしたのち頻出単語を可視化した後、文章を1月1日～3月1日、3月2日～6月1日のものに分割し、sklearnモジュールを使用しTF-IDF計算をし、出力結果をExcelファイルに書き込みどの単語が重要視されているのかなどを考察する。

なおPythonのバージョンは3.7.2、IDEはPyCarm、使用したモジュールはurlib ,untangle, MeCab, numpy, bs4, pandas, (matplotlib), sklearn ,openpyxlとした。

## **1.1. データ収集**

国会会議録APIをPythonのプログラムで操作し、データ収集したい期間:1月1日～6月1日、発言者は首相のみ、会議は予算委員会とクエリを指定し、2020.txtと言うテキストファイルに保存した。プログラム名はreturn\_data.pyとした。

## **1.2. 形態素解析と頻出単語の出力**

先程作成した2020.txtを、形態素解析器MeCabを使用して2020.txt.mecabというmecabファイルを作成し(プログラム名はmake\_mecab\_file.py)、それを元に、3文字以上の名詞を抽出し、それを元に上位の頻出単語のトップ30を出力した(プログラム名はremove&show.py)。また、不要な単語が出力されないよう京都大学が作成したストップワードを利用し、またストップワード内にない言葉をmydictとして追加した。出力結果は図1のようになる。

|  |
| --- |
| [('ホテル', 297), ('事務所', 289), ('皆さん', 280), ('ウイルス', 263), ('コロナ', 246), ('とおり', 237), ('お答え', 213), ('それぞれ', 127), ('さまざま', 122), ('我が国', 111), ('考え方', 105), ('お願い', 96), ('先ほど', 89), ('全日空', 84), ('ＰＣＲ', 81), ('いずれ', 79), ('小規模', 74), ('大統領', 66), ('イベント', 65), ('ＷＨＯ', 59), ('自衛隊', 59), ('その後', 58), ('自民党', 54), ('たくさん', 54), ('マスク', 53), ('資金繰り', 53), ('瀬戸際', 50), ('カジノ', 49), ('自治体', 48), ('リスク', 46)] |

　　図1 頻出単語TOP30(3単語以上の名詞のみ)

# **1.3. ＴＦ－ＩＤＦの実装**

**TF-IDFとは、文章中に含まれる単語の重要度を評価する手法の1つであり、TF(TermFrequency)はそれぞれの単語の分法内の頻出頻度を指し、IDF(Inverse Document Frequency)はそれぞれの単語がいくつの文書で共通して使用されているかを表している。式で表すとTF=**

**+1**

**となり、TF-IDFはTF×IDFの値で表される。**

**今回はpythonのsklearnモジュールのTfidfTransformerクラスを使用し実装した。実装方法は、全処理した2020.txtを2分割し(それぞれ前半.txt, 後半.txtとした)、期間も前者は1月1日～3月1日、後者は3月2日～6月1日とし、それぞれもmecabファイルを作成して、モジュールnumpyを利用しベクトル化した文章に計算を加え、上記の式に従って値を変更したのち、Excelに出力しデータを考察することとした。プログラム名はTF-IDF.pyとし、Excelのファイル名は出力結果.xlsxとした。また、セルのnum1は前半部分、num2は後半部分を示している**

# **4. 考察**

**結果のExcelファイルから、後半部分を表すnum2において、他にも数多くの単語について話されている中、特に新型コロナウイルス関連の単語であるウイルス・コロナ・ワクチン・PCRなどのTF-IDF値が大きいことから、それらの言葉は後半部分すなわち3月2日以降急速に議論が交わされていることがわかり、政府のコロナ対策における対応が少し遅れているということが考察される。**

# **5. むすび**

本論文では、国会会議録APIからデータを抽出した文章をTF-IDF法を実装して急上昇ワードを出力する方法を提案した。一応は形のあるものとなってはいるが、著者自身があまりTF-IDF法などの機械学習全般について理解不足な部分もある上、形態素解析における頻出単語をmatplotlibで可視化を試みたが失敗し、あえなくprint文を使いなんとか目に見える状態にすることはできたが、まだまだ改良の余地があるプログラム・論文であることが明らかとなった。

以上のことから、自然言語処理機械学習においてさらなる研究・勉強を進めていきたいと考えている。また、この場を借りて黄潤和教授、TAの下窪聖人さんにお礼を申し上げたい。

# **参　考　文　献**

1. <https://blog.aidemy.net/entry/2018/05/11/162024>
2. <https://github.com/ikegami-yukino/mecab/releases/tag/v0.996>
3. [file:///C:/Users/admin/Desktop/%E3%83%97%E3%83%AD%E3%82%B8%E3%82%A7%E3%82%AF%E3%83%882A/%E7%AC%AC%E5%9B%9B%E5%9B%9E/L4-revised.pdf](file:///C:\Users\admin\Desktop\%E3%83%97%E3%83%AD%E3%82%B8%E3%82%A7%E3%82%AF%E3%83%882A\%E7%AC%AC%E5%9B%9B%E5%9B%9E\L4-revised.pdf)(講義資料第四回のPDF)
4. <https://fresopiya.com/2019/04/09/mecabhinshi/>
5. <https://www.dskomei.com/entry/2019/04/04/191506>
6. <http://svn.sourceforge.jp/svnroot/slothlib/CSharp/Version1/SlothLib/NLP/Filter/StopWord/word/Japanese.txt>(ストップワード)
7. <https://yut.hatenablog.com/entry/20130215/1360884220>
8. <https://www.jstage.jst.go.jp/article/tjsai/24/6/24_6_488/_pdf>(イベント系列マイニングを目的とする新聞記事からの時間情報に基づく単語抽出)
9. <https://mieruca-ai.com/ai/tf-idf_okapi-bm25/>