DOI : 10.17314/jjlc.2022..61.002

トピックモデルを用いた日本語テキストマイニングの研究*

－旧JLPTの読解の既出問題に対する分析を中心に－

金 曄 泳**

<目 次>

Key Word : テキストマイニング(text mining), トピックモデル(topic model), LDA (latent dirichlet allocation), パイソン(python), ゲンシム(genshim)

1. はじめに

IT技術の発達によって人類が作り出している情報量はその予測すら遥かに超えている。2003年のカルフォルニア州立大学の研究チームの報告によると全世界が年間で生産する情報量が約54億GB(Gigabyte・ギガバイト)であっ

* 이 논문은 2021년도 동덕여자대학교 연구비 지원에 의하여 수행된 것임(연구번호: 20210156). This study was supported by the Dongduk Women's University grant(No. 20210156)

** 同徳女子大学 日本語学科 副教授, 日本語学

て、これは1900万以上の蔵書をもつアメリカの議会図書館の約50万倍にあたる莫大な情報量である[1]。しかし、2017年になるとそれを遥かに超えて、わずか一日で全世界が生産するデータ量が約25億GBにまで上り(イ・ビョンウク(이병욱), 2020:1)、さらに2025年には約4,466億GBまで上ると予測されることに至った(IDC[2], 2020)。

このような莫大な情報量の増加は必然的に情報の効率的な利用方法に対する関心に繋がるが、その中でビックデータの中で意味のある情報を抽出できる技術であるテキストマイニング(Text Mining)が注目を浴びている。本研究ではテキストマイニング、その中でもトピックモデル(Topic Model)の技法を通じて大規模のテキストデータのトピック(主題)を抽出し、トピックモデルの技法に対する紹介とそれを基盤としてトピックの性格や分布に至るまで、テキストの著者である作者の観点や意見を把握することを目標としたが、これは日本語学や日本語教育分野においても大規模のテキストデータの処理の必要性が高まっている今であるからこそ有効な研究方法として意義があると考えられる。今回は旧JLPTの読解パートの既出問題に対するトピックモデル(Topic Model)分析を行い、どのようなトピックが主に取り上げられているかを確認したい。

2. 先行研究

2.1 テキストマイニング(Text Mining)

テキストマイニングはデータマイニングの一つで、後者であるデータマイニングは精製された構造的なデータを用いて分析を行うことであるが、テキストマイニングは自然語のように精製されていない非構造的なデータ(Unstructured data)であるテキストにおける関係性やパターンなどを抽出す

1)「世界の情報生産量、年間54億GB」2006.04.04.
https://www.hankyung.com/international/article/2003110339868
2) IDC：International Data Group. https://www.idc.com

る統計的な研究方法である。つまり、テキストマイニングは数字のような定型のデータではなく、テキスト・音声など、形態と構造が一定ではない自然語の処理技術に基づいて意味のある情報を見出して、加工・分析することを目的とするのである。よってテキストマイニングは膨大な言語データの処理のために欠かせないものであると言えよう。

日本学の分野でもこのようなテキストマイニングが注目され始めていて、李敬淑他(2018)、落合由治(2020)などの研究ではテキストマイニングの研究事例などを紹介しながらその有効性を述べている。また、他の学問分野と同様、李敬淑(2021)のような論文DBにおける調査を通じて研究動向を分析した研究も見られている。続いて、李有姫(2019・2020・2021)、金嚅泳(2020)、金明哲他(2020)、金昭喜(2021)、キム・ヘヨン(김혜연、2022)などの研究では実際のテキストにおける分析を行ったが、その中でも機械的にテキストを構成する要素の頻度を集計するために専用のプログラムを開発した金明哲他(2020)と日本語能力試験の問題における分析を行った李有姫(2019・2020・2021)などが本研究に示唆する点が多い。しかし、日本学分野においてテキストマイニングに関する研究は絶対的な数が少ない上で、テキストにおける特定の単語の頻度やそれと共起する言葉との関連性を見出すことを目標とした研究が多く、従来のコーパス言語学とコロケーションの研究と差別化されにくい点が指摘される。従って本稿では、実際のテキストにおけるテキストマイニングによる頻度分析や共起情報からさらに一歩進んで、Latent Dirichlet Allocation(潜在的ディリクレ配分法、以下LDA)を用いて複数の単語の共起性によって創発される情報である文書のトピックを見出す実験を行った。

2.2 トピックモデル(Topic Model)

テキストマイニングの中でトピックモデルは構造化されていない膨大なテキストデータの中でトピックを見出すためのアルゴリズムであって、文書が生成される過程を確率を用いてモデル化した確率モデルとも言える。簡単に

いうと、特定のトピック(主題)の文書ではそのトピックと関連する単語が多く現れる。例えばトピックが「経済」である文書には「景気、株、会社」などの単語が、その一方で「旅行」がトピックである文書には「宿泊、交通、グルメ」などの単語の出現率が高くなる。つまり、トピックモデルでは共に出現する確率が高い単語の集まり(グループ)を暫定的なトピック(主題)であると定義できるが、そのようなグループを構成することがトピックモデルリングになるという概念である。

このようなピックモデルリングは大規模のテキストを扱う多様な分野の研究において分析の道具として使用され始めている。日本学の分野でも黒田絢香(2021)はピックモデルの一つであるLDAを用いて効果的なビジュアライゼーションツールの開発を、黄晨雯(2021)は新たなトピックモデルであるTop2Vecを使用してトピックという視点からの小説の解読を試みた。また、李広微他(2020)もトピックモデルを用いて現代小説における接続表現の使用変化の過程を分析した。このように日本学分野におけるトピックモデルに対する関心が高まっていて、ツールの開発から応用まで研究が広がっている。しかし、関連研究は今始まったばかりであって数が少なく、また本稿における調査対象である実際の日本語能力試験の既出問題において分析を行った研究も管見の限り見当たらなかった。

3. 研究方法

金嚅泳(2020:27)によると、テキストマイニングは大きく二つの工程に分けられるが、一番目が資料処理(Data Processing)であって、二番目が資料分析(Data analysis)である。本稿では、20年間の旧日本語能力試験(以下、旧JLPT)の既出問題を収集し、そのテキストを資料処理プロセスであるデータの前処理によって非構造化データを分析可能な形態に加工・精製した。続いて実際の分析を行ったが、具体的には以下のようになる。

3.1 データの収集

本稿では金嚅泳(2014:215-216)の基準に従って、以下の(1)のように、20年間[3]にわたる全ての旧JLPT(1~4級)の既出問題の中で読解パートのテキストのみを選別して収集した。

(1) [旧JLPTの読解・文法パートにおけるテキスト]

a. 旧JLPT・1級
1) 1990~1991・1993・1998年：I・II・III 1~5
2) 1992・1994年：I・II・III 1~7
3) 1995~1996・1999年：I・II・III 1~6
4) 1997年：I・III 1~5 (著作権で未公開であるIIを省略)
5) 2000~2001年：I・II 1~4・III 1~3
6) 2002・2004・2008・2009年1次：I・II 1~4・III 1~4
7) 2003年：I・II 1~3・III 1~4
8) 2005~2007・2009年2次：I・II 1~4・III 1~5

b. 旧JLPT・2級
1) 1990년：I・II・III 1~4
2) 1991・1999年：I・II・III 1~7
3) 1992・1995年：I・II・III 1~5
4) 1993~1994・1996~1998年：I・II・III 1~6
5) 2000年：I・II 1~3・III 1~6
6) 2001年：I・II・III 1~4
7) 2002・2004~2005・2007・2009年2次：I・II 1~3・III 1~5
8) 2003・2008・2009年1次：I・II 1~3・III 1~4
9) 2006年：I・II 1~3・III 1~3

c. 旧JLPT・3級
1) 1990~1991年：IV
2) 1992年：IV 1~4
3) 1993~1996年：V 1~4・VI
4) 1997~2009年・2次：V・VI

d. 旧JLPT・4級
1) 1990年：V 1~6

3) 1990~2009年における旧JLPTの既出問題。

2) 1991年：V 1~5
3) 1992~1993年：V 1~3
4) 1994~1995年：VI 1~3
5) 1996~1999~2001年：V･VI 1~3 金嚅泳(2014:215-217)

3.2 データの前処理

上記の3.1で収集した「*.txt」形式のテキストのローデータ(n=437)を形態素分析･統計処理などのデータ分析に適する形式に変換する前処理を行ったが、詳しくは次のようになる。まず、分析対象になるテキストフィールドから問題番号･選択肢番号･記号など、必要ではない文字列を除去或いは他のフィールドへの移動を行い、年度･出典(問題番号)･テキストの三つのフィールドを持つ「*.csv」形式に変換した。さらに全てのテキストデータをまとめて「旧JLPTの読解テキストデータセット」を構築したが、詳しくは以下の(2)と<図1>のようになる。

(2) [旧JLPTの読解パートにおけるデータセット]
a. 旧JLPT･1級：JLPT_Lv1.csv
b. 旧JLPT･2級：JLPT_Lv2.csv
c. 旧JLPT･3級：JLPT_Lv3.csv
d. 旧JLPT･4級：JLPT_Lv4.csv
e. 旧JLPT･1~4級：JLPT_All.csv

Date	No	Text
1990	GR01-1	マンションの十階の部屋を出てエレベーターを待っていたら、扉が開くやいなや、
1990	GR01-2	一般論として言えば、最近の学生にとって仕事の選択とは、「銀行員」とか「公務員」
1990	GR01-3-1	何とかしたくても、こちらに打つ手がない以上、先方の意のままにならざるをえな
1990	GR01-3-2	宇宙開発研究所がこれから先どれだけ計画を実現していくことができるかは、ひと
1990	GR01-3-3	理論上はそういうことになるとわかってても、事実の裏付けがないうちはどうして
1990	GR01-3-4	魚が周囲のいろいろなものやさまざまな生物に気がついたとして、最後になってや
1990	GR01-3-5	「理論を共有する」ということと、「理論を理解する」というのは、まったくの別問題
1991	GR01-1	死体ははたしてだれのものか。　自分のものだとしても、死んだあとでは、所有権
1991	GR01-2	私の知っている寿司屋の若い主人は、亡くなった彼の父親を、いまだに尊敬してい
1991	GR01-3-1	私は自分の行動について、いかなる批評もしなかった。人の眼(め)には、或いは、そ

<図1>「旧JLPTの読解テキストデータセット：csv」の一部

3.3 データの分析

本研究ではトピックモデル分析のためにPythonを使用して独自のプログラムを開発したが、その基本になる仕様と利用したツールとパッケージなどは以下の(3)のようになる。

(3) [データ分析ツール及び環境]

a. Language： Python – Anaconda 1.10.0 / Jupyter Notebook 6.4.12
b. 形態素分析：SudachiPy
c. Python統計パッケージ：Gensim / pyLDAvis / Seaborn / Wordcloudなど
d. OSなど：Win10 Pro / Mem 32G / Intel(R) Core(TM) i7-1065G7 / GPU Off

3.3.1 形態素分析及びBow形式

上記の(2)csvのデータセットにおけるテキストを<表2>のようにSudachiパッケージを使って形態素分析を行った後、<表3>のようにその各文書をBoW形式で保持した。それに使われたPythonコードの実装は以下の<表1>になる。BoW(Bag of Words)形式とは文書群を文書(Document)と単語(Word)の行列で表現する形式で、それぞれの要素に出現数のカウントによって値を与えたものである。

<表1> 形態素分析とBoW形式変換のPythonコード

```
# トピックモデル #1 – 形態素分析コード by Kim Yu Young
from tqdm.notebook import tqdm
from sudachipy import tokenizer
from sudachipy import dictionary
from gensim import corpora
from gensim import models
from gensim.corpora.dictionary import Dictionary
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import seaborn as sns
from collections import Counter
from wordcloud import WordCloud
```

```
import pickle
import numpy as np

class SudachiAnalizer():
    def get_token(self, source) :
        tokenizer_obj = dictionary.Dictionary().create()
        mode = tokenizer.Tokenizer.SplitMode.C
        result = tokenizer_obj.tokenize(source, mode)
        word_list = []
        for mrph in result:
            if not (mrph == ""):
                norm_word = mrph.normalized_form()
                hinsi = mrph.part_of_speech()[0]
                if hinsi in  ["名詞", "動詞", "形容詞"]:
                    word = tokenizer_obj.tokenize(norm_word, mode)[0].
dictionary_form()
                    word_list.append(word)
        return word_list
corpus = []
docs = []
PATH = ""
sudachi = SudachiAnalizer()
JLPT_Texts = []
JLPT_Links = []
JLPT_Titles = []

import pandas as pd
JLPT_text = pd.read_csv("JLPT_GR/JLPT_All.csv", encoding = 'utf-8')
JLPT_Links = JLPT_text["No"]
JLPT_Titles = JLPT_text["Date"]
JLPT_Texts = JLPT_text["Text"]

i = 0
j = 0
for JLPT_text in tqdm(JLPT_Texts):
        token = sudachi.get_token(JLPT_text)
        corpus_data = {"Title" : JLPT_Titles[j], "tag" : JLPT_Links[j], "token"
: token}
```

```
            corpus.append(corpus_data)
            docs.append(corpus_data)
            i = i + 1

with open('model/corpus.pkl', 'wb') as temp:
  pickle.dump(docs, temp)

f = open('model/corpus.pkl','rb')
docs = pickle.load(f)
tag_list = []
for doc in docs:
    tag_list.append(doc["tag"])

df = pd.DataFrame(tag_list)
tag_counts = df[0].value_counts()
tag_counts
# docsにおける走査の順番確認
tag = ""
for i in tag_list:
    if i != tag :
        print(i)
        tag = i
text_list = []
for item in tqdm(corpus):
    text_list.append(item["token"])
```

<表2> 形態素分析及び単語の抽出例 - 問題「GR01-1」の場合

GR01-1
{'Title': 1990, 'tag': 'GR01-1', 'token': ['マンション', '10', '階', '部屋', '出る', 'エレベーター', '待つ', '居る', '扉', '開く', '否', '言う', '猫', '下りる', '来る', '人', '乗る', '居る', '驚く', '猫', '見る', '居る', '猫', '慌てる', '様子', '無い', '非常階段', '方', '歩く', '行く', '仕舞う', '屋上', '方', '目指す', '居る', '遣る', '乗る', '来る', '人', '乗る', '居る', '追い出す', '無人', '扉', '開く', '時', '乗る', '仕舞う', '為る', 'ボタン', '押す', '中', '大人しい', '待つ', '居る', …… 以下省略

<表3> Bow形式の例 – 問題「GR01-1」の場合

['マンション', '10', '階', '部屋', '出る', 'エレベーター', '待つ', '居る', '扉', '開く', '否', '言う', '猫', '下りる', '来る', '人', '乗る', '居る', '驚く', '猫', '見る', '居る', '猫', '慌てる', '様子', '無い', '非常階段', '方', '歩く', '行く', '仕舞う', '屋上', '方', '目指す', '居る', '遣る', '乗る', '来る', '人', '乗る', '居る', '追い出す', '無人', '扉', '開く', '時', '乗る', '仕舞う', '為る', 'ボタン', '押す', '中', '大人しい', '待つ', '居る', '違い', '無い', '時間', '用', '済ませる', …… 以下省略

3.3.2 辞書とコーパスの作成

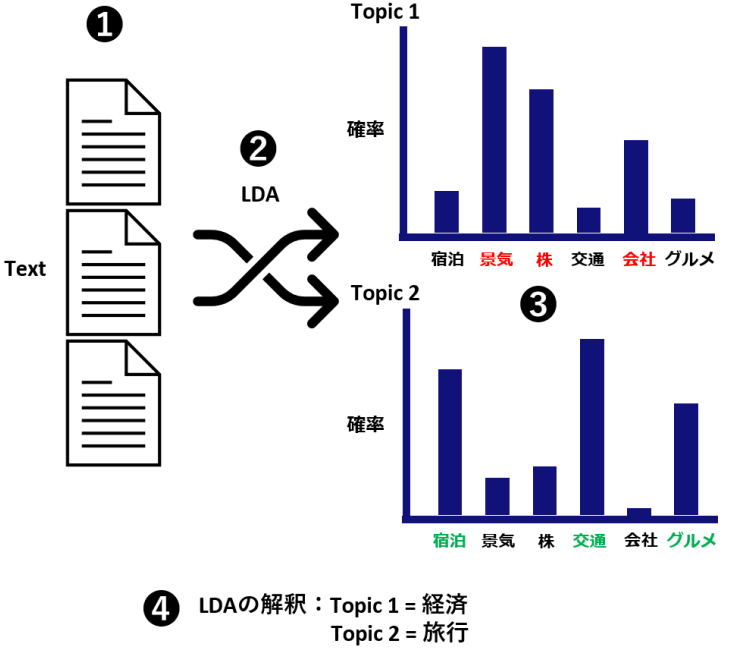
それに続いて、本稿におけるトピックモデルはLDAを仮定して考えられていたので、そのために以下の<表4>のように辞書を作成してコーパスを生成したが、具体的には計2584語を学習させた辞書が生成できた。

<表4> 文書データにおける辞書作成のPythonコード

```
dictionary =corpora.Dictionary(text_list) #辞書の作成
dictionary.filter_extremes(no_below=2,no_above=0.2)
corpus=[dictionary.doc2bow(tokens) for tokens in text_list]
```

3.3.3 LDA(Latent Dirichlet Allocation)によるトピック抽出及び可視化

LDAを実行するにあたり、LDAに関して簡単に述べてから論を進みたいと思う。2.2節で簡単に述べたようにトピックモデルリングをするに際して使われるアルゴリズムがLDAである。



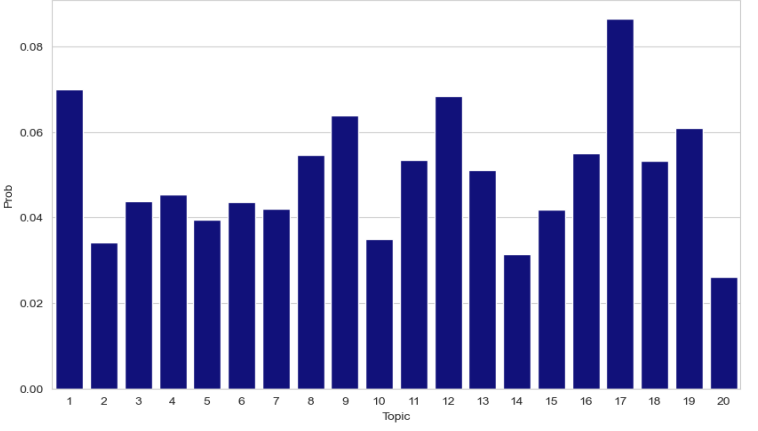
<図2> テキストにLDAを実行した例：経済と旅行のトピックを持つテキストの場合

上記の<図2>のように例えると、<図2>の❶LDAは大量の文書データ(<図2>の❶)から単語のグルーピングを行うモデルであって、このグループ一つ一つが「トピック」になる(<図2>の❸)。ここで気を付けなければならない点が二点あるが、まず「トピックの数を予め決めておかなければならない」こと、それから「どのような意味を持つグループを作るかは予め決めるのではなく、産出された確率に基づいた人間の解釈が必要になる(<図2>の❹)」ということである。ちなみに、本稿では任意のストップワードを指定せず、そのまま分析を行なった。また、適切なトピック数を４まで絞り込み、LDAを実行してトピック数を決定したが、LDAにおける詳しい流れとその結果は続いて4節で述べることにする。

4. 分析結果

4.1. LDAにおけるトピックの数の設定

本稿ではLDAによるトピックモデルリングを遂行するに際して、可能な限り恣意性を排除することを目的として「Perplexity[4]」と「Coherence[5]」をテキスト種類毎に計算した結果から、トピック数を検討した。まず仮にトピックを20に設定してLDAを行ったが、その結果であるトピックの分布とトピック間の相関は以下の<図3>と<図4>、<表5>になる。



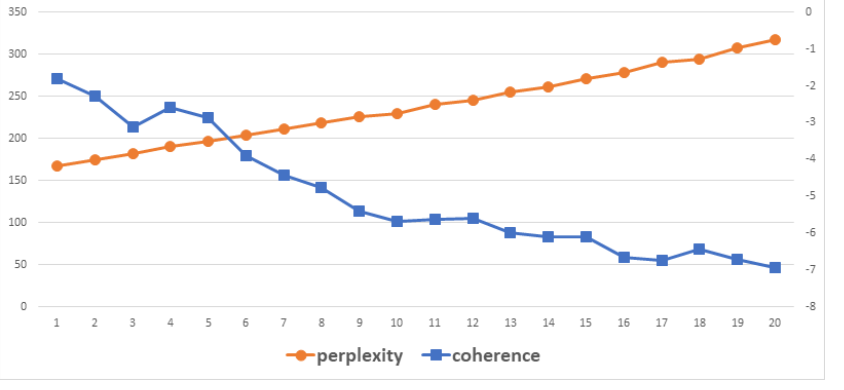
<図3> 旧JLPT読解のトピック分布(トピック数20の場合)

4) 平均分岐数：モデルの予測精度の評価指標、汎用能力を表す指標。一般的にトピック数を増やすと共に低くなる傾向にあり，低い程良い。

5) トピックの品質を測る評価指標、抽出されたトピックが人間にとって解釈しやすいかどうかを表す。トピック全体のCoherenceが高ければ、良いモデルと考えられる。

<表5> トピックの数20の設定における単語の所属確率–Topic1と2の場合

TOPIC 1			TOPIC 2		
[	word	prob	[	word	prob
0	子供	0.013841	0	年	0.017454
1	仕事	0.010479	1	相手	0.009418
2	得る	0.007590	2	親	0.008853
3	家族	0.007430	3	機械	0.008444
4	時間	0.006660	4	日	0.008399
…	…	…	…	…	…
995	捲る	0.000214	995	溢れる	0.000103
996	なくなる	0.000213	996	会社	0.000102
997	信じる	0.000213	997	笑う	0.000102
998	溜め息	0.000213	998	展開	0.000101
999	親子	0.000211	999	きつい	0.000100



<図4> トピックの数20の設定における「perplexity」と「coherence」

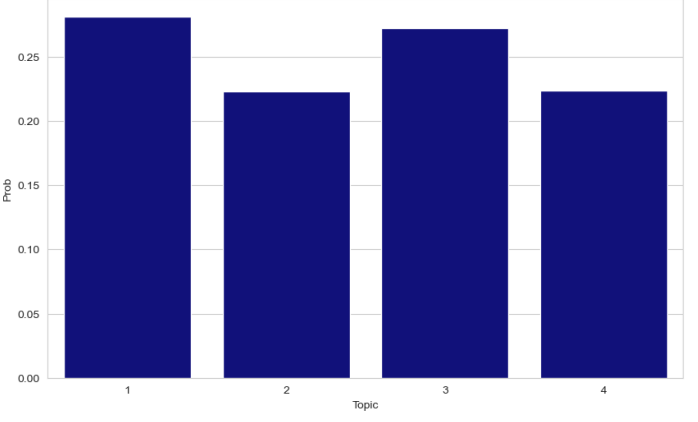
その中でBoWコーパスで学習を行った結果、perplexityとcoherenceは上の<図4>のようになった。脚注4･5のよう、perplexityは低ければ低い程、coherenceは高ければ高い程良いモデルになる。従って、今回のデータについては数を「4」に設定した。

4.2. LDAによるトピックモデルリング

前節で述べたように、本稿ではテキストにおけるトピックの数を「4」に設定してpyLDAvisパッケージによるLDAを遂行したが、そのトピックモデルにおけるトピックの出現率とトピック別の距離や単語分布結果は以下の<図5>と<図6>のようになる。

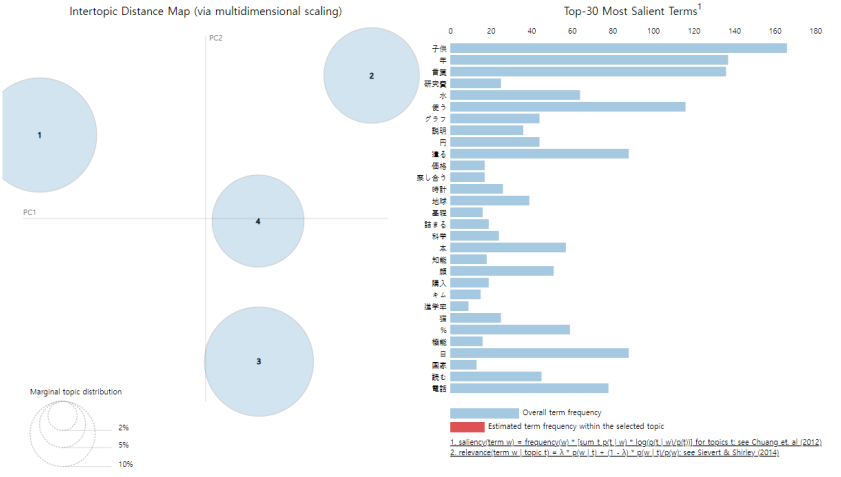
まず、日本語能力試験(JLPT)は文字通りに受験者の日本語の能力を測定するための言語検定試験であって、そのテキストの選定や制作など、問題の出題に際してトピックの偏りはあまり望ましくないと思われる。そのような主題者(作者)の意図は本稿の考察でも確認できるが、以下の<図5>のようにそのトピック別出現確率は比較的に偏っていないことが分かる。

続いて、以下の<図6>の左側における四つのサークル同士の距離はトピック同士における相違の度合いを示すものであって、本モデルにおける四つのトピックがお互い区分できるトピックであることが確認できる。具体的には脚注6を参考にすること[6)]。



<図5> 旧JLPT読解のトピック分布(トピック数4の場合)

6) LDA可視化：本稿における四つのトピック同士の距離やトピック別単語の分布は以下のリンクから数字入力やマウス・オーバー或いはクリックによって実際に確認できる。http://japanese.or.kr/pyldavis_output_pcoa_4.html



<図6> 旧JLPT読解のトピックのPCoA(Principal Coordinate Analysis)

4.3. トピック分析

本節ではそれぞれグループ化した四つのトピックの解釈を行うが、以下の<表6>の❶トピック1から確認してみると、「子供」「時間」「言葉」「仕事」「家族」「電話」「聞く」「話」「得る」「与える」「問題」などの主要単語から❶トピック1は「家族と仕事などの私的関係」に関わるテキストだと解釈して良いと思われる。続いて、❷トピック2の場合、「年」「遣る」「日」「時間」「方」「出る」「相手」「説明」「前」「下さる」などの主要単語から、「日程にまつわるコミュニケーション」に関わるテキストであると解釈した。また、❸トピック3の場合、「言葉」「人間」「訳」「知る」「世界」「日本」「相手」「社会」などの主要単語から「国や社会にまつわる公的関係や場面」に関わるテキストで、最後に❹トピック4の場合「使う」「円」「生活」「食べる」「読む」「部屋」「必要」「買う」「研究費」などの主要単語から、「経済活動」に関わるテキストであると解釈した。

このような点から、旧JLPTの読解パートのテキストは、多様なトピックが偏らず選ばれてはいるが(4.2節)、そのトピックは主に「家族と仕事などの私的関係」「日程にまつわるコミュニケーション」「国や社会にまつわる公的関

係や場面」「経済活動」に分けられることが確認できる。これはJLPTのテキストにおけるトピックモデルリングが指導に当たる先生や受験する学習者の両方において出題傾向の分析に役立つと考えられる。

<表6> 四つのトピックにおけるワードクラウド

No	ワードクラウド
❶ Topic 1	
❷ Topic 2	
❸ Topic 3	
❹ Topic 4	

5. おわりに

本稿では現在膨大化しているテキストデータの効率的な活用のための試みの一つとして、トピックモデルというテキストマイニングの技法を日本学分野にも取り入れた。具体的には過去20年間の全ての旧JLPTの読解パートの既出問題のテキストを収集してトピックモデル分析を行ったが、そのような考察によって次のような点が明らかになった。まず、旧JLPTの出題者は問題におけるテキストの選定や制作に際してトピック(主題)別偏りを避けようと努力していたことが実際のデータによって確認できた。続いて、同テキストは統計的に主に四つのトピックに分類できる上で、それぞれ「家族と仕事などの私的関係」「日程にまつわるコミュニケーション」「国や社会にまつわる公的関係」「経済活動」のような内容になることが確認できた。

このような本稿の考察におけるトピックモデル分析の技法とその結果は実際の既出問題に対する実証的な分析であった点、そして日本語の教育分野のみならず、大規模な日本語のテキスト処理が必要とされる日本学のあらゆる分野に応用できると思われる点で意義があると考えられる。もちろん、本稿における考察は新JLPTではなく旧JLPTのテキストに限定されている点、過去20年間の全てのデータではあるがそのデータの量が比較的に大きくない点、最後に本文の制約により他のテキストとの比較対照分析が出来なかった点などでは改善の余地があると思われるが、今後の課題としたい。

【参考文献】

金明哲・鄭弯弯(2020)「テキストコーパスマイニングツールMTMineR」『計量国語学』32-5, 計量国語学会, pp.265-276

金昭喜(2021)「テキストマイニングを活用した「X{まで}」構文の語彙分析―「X{さえ}」「X{も}」との比較・対照を中心に―」『日本学報』126, pp.121-143

金嚅泳(2020)「テキストマイニングを用いた日本のメディアの韓国ニュースにおける

感情の推移に対する分析-Pythonを用いた「単語感情極性対応表」の分析を活用して-」『日本語学研究』65, 韓国日本語学会, pp.25-43

キム・ヘヨン(김혜연)(2022)「텍스트마이닝을 활용한 한일 어휘·문화 교육의 가능성—'トイレ', '화장실' 관련 어휘 및 문화를 중심으로—」『日本語文学』92, 韓国日本語文学会, pp.139-159

李広微・金明哲(2020)「トピックモデルに基づいた現代小説の接続表現の分析」『日本行動計量学会大会抄録集』48, pp.152-153

李敬淑・郭乃貞(2018)「ICT時代の日本語研究と教育におけるテキストマイニングの有効性」『韓国日本語教育学会・学術発表論文集』2018巻0号, 韓国日本語教育学会, pp.65-69

李敬淑(2021)「텍스트마이닝 기법을 활용한 일본어 학습자의 음성에 관한 연구 동향 분석」『日本語学研究』69, 韓国日本語学会, pp.93-106

李有姬(2019)「텍스트 마이닝을 활용한 일본어능력시험 내용 연구—JLPT N3 문자·어휘를 중심으로—」『日本文化学報』82, 韓国日本文化学会, pp.5-25

_____(2020)「텍스트 마이닝을 활용한 일본어능력시험 내용 연구—JLPT 1급 문자·어휘를 중심으로—」『日本語文學』90, 日本語文学会, pp.155-179

_____(2021)「텍스트 마이닝 기법을 통한 일본어능력시험 분석과 학습 교육—JLPT 4급 문자·어휘를 중심으로—」『日本文化學報』91, 韓国日本文化学会, pp.283-304

柳銀珠・吳相勛・朴時四(2014)「日本の高齢者の「観光」意識に関するテキストマイニング分析」『日本近代学研究』46, 韓国日本近代学会, pp.371-386

黄晨雯(2021)「Top2Vec による小説の探索的研究：程小青の作品解読を中心に」『言語文化共同研究プロジェクト』2020, 大阪大学大学院言語文化研究科, pp.43-53

落合由治(2020)「AI技術からみた日本語学, 日本語教育研究の展望と課題—日本語教育の繋がりと協働の新領域をめざして—」『日本語教育研究』50, 韓国日本語教育学会, pp.23-34

黒田絢香(2021)「トピックモデル可視化ツールの開発に向けて」『言語文化共同研究プロジェクト』2021, 大阪大学大学院言語文化研究科, pp.5-13

藤田郁(2022)「LDAトピックモデルによるIPAテキスト分析の試み：アルフレッド・テニスンの韻文を用いて」『言語文化共同研究プロジェクト』2021, 大阪大学大学院言語文化研究科, pp.15-38

David M. Blei, Andrew Y. Ng, Michael I. Jordan(2003), Latent Dirichlet Allocation, *Journal of Machine Learning Research* 3, the MIT Press, 993-1022

「AJ」- All about Japanese Study
http://www.japanese.or.kr(検査日: 2022.08.01)

SudachiPy v0.5.4 https://github.com/WorksApplications/SudachiPy
(検査日: 2022.08.01)
週刊朝鮮「매일 20조비트…코로나19가 앞당긴 '정보 재앙' 큰 일」
http://weekly.chosun.com/news/articleView.html?idxno=16161(検査日: 2022.08.01)

◇논문마감: 2022. 11. 20.
◇심사개시: 2022. 11. 24.
◇게재확정: 2022. 12. 03.

〈Abstract〉

Research on Japanese text mining using the Topic Model
－Focusing on the analysis of past test reading comprehension questions in the previous format of JLPT－

Kim, YuYoung

In this paper, as one of the attempts to effectively utilize the vast amount of text data, I have introduced a text mining technique called Topic Model into the field of Japanese studies. Concretely, the texts of the reading comprehension parts of the previous format JLPT for the past 20 years were collected, and Topic Model analysis was carried out. The following points were made clear by such a study. First of all, it was confirmed from actual data that the subjects of the previous format JLPT tried to avoid topic-specific biases when selecting and producing the texts for the questions. Next, the text can be statistically classified into four main topics: “Private relationships such as family and work,” “Communications related to schedules,” “Public relations related to the country and society,” and “Economic activity.”

The techniques and results of topic model analysis in this paper were empirical analyzes of actual existing questions. It is considered significant in that it can be applied to all fields of Japanese studies that are needed. Of course, the discussion in this paper is limited to the texts of the previous format JLPT, not the new format JLPT, and the amount of data is relatively small, although it covers all the data for the past 20 years. In addition, a comparative analysis with other texts was not possible. Therefore, it seems that there is still room for improvement in this paper, but I would like to address this as a future issue.