Validating Python Packages for Text Analysis in Language Research

Sumi Han & Minji Kim

(Hallym University)

The 2023 Linguistics Society of Korea
December 9, 2023

Hallym University

Table of Contents

Hallym University

Research Background

Hallym University

Need of Research

This work is part of a research team project on "*A Convergence Study for Deep-Learning Based AI Fiction Generation with Human in the Loop.*" (funded by National Research Foundation of Korea)

How to identify linguistic patterns in human-authored AI-authored novels?

Need of Research

- Text metrics or analytics have long been used in fields such as the digital humanities to **understand and compare text corpora** (Hansen, Olsen, & Enevoldsen, 2023)

- User-friendly computer analytic tools, such as Python libraries, needed for the **automatic examination of extensive language datasets, reducing time and effort, and reproducibility** (Albrecht, Ramachandran, & Winkler, 2020).

- Many of them are not well-known in the linguistic community, and their validity is seldom assessed.

  An in-depth investigation on various text analytics libraries or packages is needed!

Why Python?

- A general-purpose, high-level programming language which is widely used in recent times and **flexibility, readability, and high level of abstraction for enhancing user productivity** (Gholizadeh, 2022; Srinath, 2017; Srinivasa, 2018).

Key Terms

Function, Module, Package, Library

- Function:
- Module: A file containing Python definitions and statements; module name.py
- Package: A collection of modules
- Library: Similar to Package

Source from https://docs.python.org/3/tutorial/index.html

Aims of Study

1. To review and validate popular Python packages for literary text analysis

2. To offer recommendations for future linguistic research

Hallym University

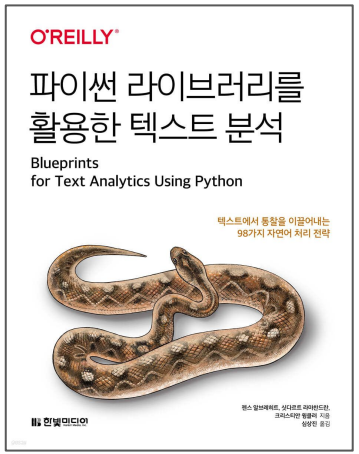
Method

- Search Process
- Data Collection & Analysis

Hallym University

Search Process

- Searched for various Python packages for text analysis on github, google scholar, etc.
- Referred to Python programming books on text analytics such as ‘Blueprints for Text Analytics Using Python’
- Search words: python, text analysis, text analytics, linguistic feature, readability, complexity, package, library, etc.
- Search Period: September 26, 2023 ~ October 10, 2023



Data Collection & Analysis

- Referred to the github statistics (e.g., star, folk, etc.) and references
- Obtained 9 packages and targeted 5 packages for analysis:

  TextDescriptives, textstat, textacy, textcomplexity, Language Feature Toolkit (LFTK)

- Collected and examined the key linguistic features of each package with its github and PyPI page and reference(s). (Google spreadsheet for linguistic features for each package)
- Compared the packages and identified strengths and weaknesses

Results & Discussion

Hallym University

Python Packages for Text Analysis

Package	(first) Release	Github Page Stars/Folks/Contributors (as of December 9, 2023)	Key Components/ Linguistic Features
textstat	June 2014	https://github.com/textstat/textstat 1K/153/37	2 components with 44 linguistic features: basics, readability
textacy	April 2016	https://github.com/chartbeat-labs/textacy 2.1K/255/30	4 components with 32 linguistic features: basics, counts, diversity, readability
TextDescritives	July 2021	https://github.com/HLasse/TextDescriptives 245/20/12	6 components with 69 linguistic features: descriptive_stats, readability, coherence, dependency_distance, pos_proportions, and quality

Python Packages for Text Analysis

Package	(first) Release date	Github Page Stars/Folks/Contributors (as of December 9, 2023)	Key Components/ Linguistic Features
textcomplexity	October 2020	https://github.com/tsproisl/textcomplexity 68/12/2	5 components with 64 linguistic features: surface, sentence, pos, dependency, constituency
LFTK (Language Feature Toolkit)	March 2023	https://github.com/brucewlee/lftk 81/26/2	4 components with 220 linguistic features: lexico-semantics, syntax, discourse, and surface

Python Packages & Linguistic Features: Google Spreadsheet

Common Features

- Mostly based on spaCy pipeline components and extensions
- A variety of linguistic features from basic descriptive statistics to readability
- A few include linguistic features such as **dependency and constituency** (e.g., **textcomplexity**); Some perform **text mining and other NLP tasks** such as entity recognition (e.g., **textaCy, LFTK**)

2.0 spaCy

Commercial open-source software or a library for advanced Natural Language Processing in Python and Cython

Github

- https://github.com/explosion/spaCy https://spacy.io/usage/linguistic-features
- https://spacy.io/usage/spacy-101
- https://spacy.io/usage/linguistic-features

Key characteristics

tokenization, lemmatization, tagging, parsing, text classification, neural network modeling, named entity recognition, text classification and multi-task learning with pretrained transformers like BERT, etc.

Reference: Honnibal, M,. Montani, I., Van Landeghem, S., & Boyd, A. (2020). spaCy: Industrial-Strength Natural Language Processing in Python. https://doi.org/10.5281/zenodo.1212303.

2.1 Textstat

Calculating statistics from text such as readability, complexity, and grade level

Github & Source Code

- https://github.com/textstat/textstat
- https://github.com/textstat/textstat/blob/main/textstat/textstat.py

Key characteristics

- 8 languages (for some features)
- **2 components with 44 linguistic features:** Basic Stats, Readability

Reference: Wikipedia

2.2 TextaCy

A Python Library that specializes in a wide range of natural language processing (NLP) such as tokenization, part-of-speech tagging, and dependency parsing

Github & Source Code

- https://textacy.readthedocs.io/en/latest/
- https://textacy.readthedocs.io/en/latest/walkthrough.html
- https://github.com/chartbeat-labs/textacy/tree/main/src/textacy/text_stats

Key characteristics

- Performing various NLP tasks, Topic modeling, and text analysis
- **4 components with 32 linguistic features**: Basics, Counts, Diversity, Readability
- Based on previous research; well-documented (See the source code)

2.3 TextDescriptives

A Python library for calculating a large variety of metrics

Github & Source Code

- https://github.com/HLasse/TextDescriptives
- TextDescriptives/src/textdescriptives/components at main · HLasse/TextDescriptives (github.com)

Key characteristics

- **7 components with 69 linguistic features**: descriptive_stats, readability, dependency_distance, pos_proportions, information theory, coherence, and quality

Reference: Hansen, L., Olsen, L. R., & Enevoldsen, K. (2023). TextDescriptives: A Python package for calculating a large variety of metrics from text. *Journal of Open Source Software, 8*(84), 5153.

Output (5 SF novels: C001 ~ C005)

textaCy

	Index Number	flesch_reading_ease	n_tokens	n_unique_tokens	n_sentences	avg_sentence_length
0	C001	83.948876	5964.0	1739.0	374.0	15.946524
	C002	83.074937	5831.0	1947.0	383.0	15.224543

TextDescriptive

	Index Number	flesch_reading_ease	n_tokens	n_unique_tokens	n_sentences	avg_sentence_length
0	C001	83.968975	5970	1736	374	15.962567
		83.087621	5840	1941	383	15.248042
		96.812400	6240	1496	571	10.928196
3	C004	89.240811	5962	1692	46	
4	C005	90.810078	5965	1553	47	

texts
(no function for type count)

	C003	93.54	6050.0	522.0	11.6
	C004	83.76	5808.0	450.0	12.9
4	C005	84.17	5747.0	458.0	12.5

For tokenization:
check stopwords, punctuations, whitespaces, contractions, etc.

2.4 Textcomplexity

A Python library for assessing **the linguistic and stylistic complexity of (literary) texts** (a special Input file needed)

Github & Source Code

- https://github.com/tsproisl/textcomplexity
- textcomplexity/textcomplexity at master · tsproisl/textcomplexity (github.com)

Key characteristics

- English, German
- **5 components with 64 linguistic features**: surface-based, sentence-based, pos-based, dependency-based and constituency-based measures
- **Core measures of lexical complexity**: Variability (TTR), Evenness (normalized entropy), Rarity (rare words), Dispersion (Gini-based dispersion), Lexical density (# of content words), Surprise (unexpected word choices), Disparity (semantically dissimilar words)

2.5 Linguistic Feature Tool Kit (LFTK)

A Python research package for extracting 220 handcrafted features (e.g. number of words per sentence, Flesch-Kincaid Readability Score) that are commonly used in computational linguistics (and language assessment/analysis)

Github & Source Code

- https://github.com/brucewlee/lftk
- List of linguistic features: Google Spreadsheet

Key characteristics

- **4 components with 220 linguistic features**: lexico-semantics, syntax, discourse, and surface
- English, General

Reference: Lee, B. W., & Lee, J. (2023). LFTK: Handcrafted Features in Computational Linguistics. In Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023) (pp. 1-19). Toronto, Canada: Association for Computational Linguistics.

2.5 Linguistic Feature Tool Kit (LFTK)

From basic descriptive stats to reading assessment measures

	Index Number	flesch_reading_ease	n_tokens	n_unique_tokens	n_sentences	avg_sentence_length	n_entities	readting_time_average
0	C001	84.208	6113.0	1536.0	374.0	16.344920	273.0	25.471
1	C002	81.378	5936.0	1779.0	383.0	15.498695	238.0	24.733
2	C003	102.156	6441.0	1247.0	571.0	11.280210	180.0	26.837
3	C004	88.254	6086.0	1476.0	467.0	13.032120	236.0	25.358
4	C005	88.857	5990.0	1311.0	475.0	12.610526	204.0	24.958

Conclusion & Implications

Hallym University

Conclusion

Purpose of this study: What linguistic features to use for analyzing literary texts?

Summary and Conclusion

- Examined the source code and reference of each Python package for text analysis
- Packages except textstat were based on previous research
- Mostly targeted basic descriptive statistics, diversity (or complexity), and readability

Easiness: textstat > TextDescriptives > textacy > LFTK > textcomplexity

Usefulness for research: LFTK > textcomplexity > textacy > TextDescriptives > textstat

Implications

Implications for Follow-up Research

- Target a set of features among similar measures (correlation analysis)
- Focus on linguistic features that represent genre, style, diversity, coherence, cohesion, etc. (Biber & Conrad, 2009)
- Make a Python package for analyzing literary texts

References

Albrecht, J., Ramachandran, S., & Winkler, C. (2020). *Blueprints for text analytics using Python*. O'Reilly Media, Inc.

Biber, D., & Conrad, S. (2009). *Register, genre, and style*. Cambridge: Cambridge University Press.

DeWilde, B. (2021). *Textacy: NLP, before and after spaCy (Version 0.12.0)*. https://github.com/chartbeat-labs/textacy

Hansen, L., Olsen, L. R., & Enevoldsen, K. (2023). TextDescriptives: A Python package for calculating a large variety of metrics from text. *Journal of Open Source Software*, *8*(84), 5153. https://joss.theoj.org/papers/10.21105/joss.05153

Honnibal, M,. Montani, I., Van Landeghem, S., & Boyd, A. (2020). *spaCy: Industrial-strength Natural Language Processing in Python*. https://doi.org/10.5281/zenodo.1212303

Lee, B. W., & Lee, J. (2023). LFTK: Handcrafted features in Computational Linguistics. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)* (pp. 1-19). Toronto, Canada: Association for Computational Linguistics. https://aclanthology.org/2023.bea-1.1

References

Srinath, K.R. (2017). Python—The fastest growing programming language. *International Research Journal of Engineering and Technology (IRJET), 4*, 354-357.

Srinivasa-Desikan, B. (2018). *Natural language processing and computational linguistics: A practical guide to text analysis with Python, Gensim, SpaCy and Keras*. Packt Publishing Ltd., Birmingham.

Ward, A. (2022). *Textstat.* Textstat. https://github.com/textstat/textstat.

Thank you!

Materials available on _____

contact: sumihan20@gmail.com

TRUNAJO
Spacy-readability
Readability
TextBlob