Tashaphyne: A Python package for Arabic Light Stemming

Taha Zerrouki

1 Bouira University, Bouira, Algeria ¶ Corresponding author

Summary

Stemming is an important task in natural language processing that involves reducing a word to its root form, or stem. In many cases, stemming can significantly improve the accuracy and efficiency of text analysis tasks such as information retrieval, text classification, and sentiment analysis. For the Arabic language, which has a rich morphology with a large number of prefixes and suffixes, stemming is particularly challenging. Tashaphyne provides an effective solution to this challenge, making it a valuable tool for researchers and practitioners working with Arabic text data.

Tashaphyne is a Python package that provides a comprehensive light stemmer and segmentor for the Arabic language. It stands out among other stemmers for its ability to perform stemming and root extraction simultaneously, unlike the Khoja stemmer, ISRI stemmer, Assem stemmer, and Farasa stemmer. Tashaphyne uses a modified finite state automaton that generates all possible segmentations, making it an extremely flexible tool for customizing stemmers without changing the code. Furthermore, Tashaphyne comes with default prefixes and suffixes, and allows for the use of customized lists to handle more complex aspects of stemming. Overall, Tashaphyne is an important contribution to the open-source community for Arabic language processing.

Statement of need

The Arabic language has a complex morphology with a rich system of prefixes, suffixes, and infixes. As a result, stemming Arabic text is a challenging task that requires specialized tools. While there are several Arabic stemmers available, they often have limitations in terms of accuracy and flexibility. Tashaphyne addresses these limitations by providing a comprehensive light stemmer and segmentor that performs stemming and root extraction simultaneously, generating all possible segmentations.

Tashaphyne is a light stemmer and segmentor in Arabic. It mostly supports light stemming (the removal of prefixes and suffixes) and provides all conceivable segmentations. Tahsphyne is a stem-based finite state automaton that extracts affixes (prefixes and suffixes) from a predefined list. It extracts and provides all possible affixations and configurations that result from a given word. Unlike the Khoja stemmer (Khoja & Garside, 1999) ISRI stemmer (Taghva et al., 2005), Assem stemmer (Chelli, 2019), and Farasa stemmer (Darwish & Mubarak, 2016), it can do both stemming and root extraction.

Tashaphyne also supports modifiable prefixes and suffixes, making it a highly adaptable tool for building customized stemmers without altering the code in any way.

Tashaphyne can be found at PyPi.org index ¹, it’s available as a demo on Mishkal, choose Tools/Analysis and as source code on Github.

¹https://pypi.org/project/tashaphyne/
Tashaphyne contains two important submodules: stemming and normalizing. Normalizing text is an important preprocessing step in natural language processing that involves transforming text data into a standardized format. Normalization of Arabic text involves several sub-tasks, including removing diacritics (Zerrouki, 2023), normalizing characters, and removing ligatures. These sub-tasks are essential for improving the accuracy of downstream tasks such as text classification, named entity recognition, and sentiment analysis. Tashaphyne, with its ability to perform light stemming and segmenting, can also assist in normalizing Arabic text, further highlighting its importance in Arabic language processing.

Tashaphyne has been developed within “Adawat”, an open-source framework for processing Arabic texts developed as part of a PhD research project (Zerrouki, 2020). Adawat includes several tools, including Mishkal (Zerrouki, 2022a) for restoring Arabic text diacritics and Qalsadi (Zerrouki, 2022b) for Arabic morphology analysis, both of which rely on Tashaphyne’s functionalities. In another project, we worked on applying the stemming algorithm to tackle the information retrieval problem in medical documents. (Al-Khatib et al., 2021).

Another framework that has incorporated Tashaphyne is the Classical Language Toolkit (CLTK (Johnson, 2014)), which provides natural language processing support for ancient, classical, and medieval Eurasian languages. CLTK uses Tashaphyne for several tasks, including corpus importer, tokenization, text conversion, and transliteration for classical Arabic (Johnson, 2014) (like the orthography of the Quran).

The SAFAR framework, a comprehensive toolkit for Arabic natural language processing, has also incorporated Tashaphyne as part of its stemmers. However, as SAFAR (Y. Jaafar & Bouzoubaa, 2015) is written in Java, Tashaphyne was translated to the Java programming language to enable its integration into the framework.

Tashaphyne is a powerful Python package designed to facilitate natural language processing tasks, with a particular focus on Arabic text preprocessing. Its numerous features make it a valuable tool for researchers and developers alike. Tashaphyne provides support for light stemming of Arabic words, root extraction, and word segmentation. It also includes a default list of Arabic affixes and allows users to customize their own stemmer options and data. Furthermore, Tashaphyne supports data-independent stemming, making it highly versatile and adaptable to a wide range of use cases.

In terms of applications, Tashaphyne is ideal for stemming Arabic text, which is a crucial step in many natural language processing tasks. It is also useful for text classification and categorization, sentiment analysis, and named entity recognition. Tashaphyne has already been used in numerous scientific publications, demonstrating its reliability and effectiveness in a variety of real-world applications. With its comprehensive set of features and wide range of potential applications, Tashaphyne is an indispensable tool for anyone working with Arabic text data.

**Mention**

Tashaphyne has been widely used as a tool in various natural language processing tasks by researchers. Stemming development and evaluation have been explored by (Atoum & Nouman, 2019; Dahab et al., 2015; EIDefrawy et al., 2015b, 2016; Younes Jaafar et al., 2017; Y. Jaafar & Bouzoubaa, 2015). Root extraction and evaluation were studied by (EIDefrawy et al., 2015a, 2017).

Tashaphyne has been utilized for text categorization (Hussein et al., 2016; Sallam et al., 2016), classification (Y. A. Alhaj et al., 2019; El Mahdaouy et al., 2016; Gharbat et al., 2019; Hijazi et al., 2022; Muaad et al., 2022; Naji et al., 2017), topic segmentation (Alahmadi et al., 2022; Naili et al., 2018), and summarization (AIoudah et al., 2019; Etaïwi & Awajan, 2022; Tanfouri & Jarray, 2022).


---


---
It has been applied to social media analysis (Almuqhim, 2016; Ameur et al., 2023; Bulbul et al., 2018; Kumar et al., 2013; Kumar, 2015), sentiment analysis (AlAyyoub et al., 2018; Saud Saleh A. Alotaibi, 2015; Saud S. Alotaibi & Anderson, 2016; Alqahtani et al., 2023; AlTwairesh et al., 2014; AlYasiri & Al-Azawei, 2019; Mouaad et al., 2023; Oraby et al., 2013; Oussous et al., 2019, 2020; A. M. Shoukry, 2013; A. Shoukry & Rafea, 2012), and tweet classification (E. Abozinadah, 2017; E. A. Abozinadah & Jones Jr, 2016; F. Alhaj et al., 2022; Brahimi et al., 2016; Mourad et al., 2017).

Tashaphyne has also been utilized for building resources such as corpora (Kuppevelt et al., 2018) and ontologies (Albukhaitan et al., 2017), question answering (Abdul Salam, 2022; Ezzeldin, 2014; Ezzeldin et al., 2015), and information retrieval (Al-Khatib et al., 2021; Mortaja, 2017; S & R, 2022).

Acknowledgements

We gratefully acknowledge the contributions of Tashaphyne light stemmer, and Arabeyes.org during the project’s inception.

References


