*Advances in Social Science, Education and Humanities Research, volume 612*
**International Seminar on Language, Education, and Culture (ISoLEC 2021)**

# How to Lemmatize German Words with NLP-Spacy Lemmatizer?

M. Kharis[1,*], Kisyani[2], Suhartono[3], Udjang Pairin[4], Darni[5]

[1, 2, 3, 4 5]*Universitas Negeri Surabaya, Surabaya, Indonesia*
[*]*Corresponding author. Email: mkharis.19010@mhs.unesa.ac.id*

**ABSTRACT**
Simple algorithms for the lemmatization process have been developed to recognize changes in a word as a result of grammatical processes and changes. Lemmatizer tools can analyze the types of word changes in the German language. Thus, this paper aims at investigating how the lemmatization of German words is aided by the Lemmatizer software. NLP Lemmatizer spacy, in cooperation with Python and Visual Studio Code, is utilized to find out the primary form of the word changes in German language. Based on the lemmatization analysis results, Lemmatizer SpaCy can analyze the shape of token, lemma, and PoS-tag of words in German. However, there are some errors identified during the process of finding out the word changes in German language.

***Keywords:*** *SpaCy, German lemmatization, lemmatize, Lemmatizer*

## 1. INTRODUCTION

Lemmatization is the process of getting the basic form of a word or might be referred as lemma of a word from its inflection form (Perera & Witte, 2005). German language is characterized having morphologically complex language that its lemmatization process using software can only be done through unique algorithms. For example, in German, there are seven changes in nouns through the suffixation process, namely *-s*, *-es*, *-e*, *-n*, *-er*, and *-ern* and vowel changes due to the addition of *Umlaut*. These suffix and vowel changes are influenced by sex (gender), number (singular or plural), and case (nominative, accusative, dative, and genitive). The aforementioned changes can be seen in the following words of *Wort, Satz,* and *Sprache*:

- *Wort: Wort, Wortes, Wörter, Wörtern*

- *Satz: Satz, Satzes, Sätze, Sätzen*

- *Sprache: Sprache, Sprachen*

The lemmatization process in these words can be done by reducing suffixes or other changes by analyzing the word level or its morphological process. Meanwhile, verbs also experience changes in form because verbs in German are flexible. This means that the verb will change its shape according to the actor's subject and its tenses. For example, the word *sprechen*, which means to 'speak' in the present tense, changes to *spreche, sprichst, spricht, sprecht, sprechen*, and changes to *sprach, sprachst, spracht, sprachen, gesprochen* in other tenses.

Reverting words that have changed their form to their basic forms helps the computer to recognize their meaning. For example, this reverting word can be used for machine translation and other machines related to computational linguistics. In general, the method for automatic or semi-automatic recognition and processing of human language with computers is called Natural Language Processing (henceforth is NLP), which is another term referring to computational linguistics.

Simple lemmatization processes have been developed to recognize the words' changes due to grammatical functions, and is called as a Lemmatizer. It works by cutting the suffixes and marking other changes by considering morphological features to find the word's primary form. Based on the introductory section's description, this paper focuses on answering the question how the users can lemmatize the German words aided by software and how the computer can provide information about the result of the lemmatization. By knowing how lemmatizer works, we can improve software performance in the fields of computational linguistics, for example: improving the quality of machine translation, text to speech or speech to text machine, speech recognition, and other language processes. In this paper, the lemmatization process employs the SpaCy software in

collaboration with Python and Visual Studio Code (VSC).

## 2. LEMMA

The Big Indonesian Dictionary on https://kbbi.kemdikbud.go.id page defines lemma as input words or phrases in the dictionary beyond the definition or other explanation given in the entry. Meanwhile, the online lexico.com dictionary defines an entry as a word or phrase defined in a dictionary or entered in a word list. According to [1] lemma is *'everything preceding the first explanation (or sense number) in a dictionary entry*' (leaving headword and word entry to retain their present meaning). From these definitions, it can be concluded that a lemma is a root of a word or phrase that is defined in a dictionary or included in a word list, apart from other explanations. In the dictionary, a lemma is in front of the explanation. The term lemma refers to the meaning of the synonym with the headword. Based on the type, the Ministry of Education and Culture divided lemmas into basic words, derivative words, rephrases, compound words, phrases, figures of speech, expressions, proverbs, acronyms, and abbreviations [2]

In English, the words *house* and *houses* are considered in different types and tokens, but these types are categorized as the same word or they so-called lemma. Thus, a lemma is the headword, its inflection, and its reduction form [3]. In general, in English, there are 8 (eight) forms of the lemma, namely *plural; third-person singular present tense; past tense; past participle; -ing; comparative; superlative; possessive*. Meanwhile, there are seven forms of the lemma in German, namely *singular-plural, third-person singular present and past tense, past participle, comparative, superlative*. These changes in conditions are called a derivation.

In German, the derivation process consists of three, namely (1) a change in construction followed by a shift in word class, (2) a modification of construction that is not followed by a shift in word class; verbs experiencing the derivations in this group, adjectives and article; (3) changes in the form of words, but not followed by changes in sound. In German, for example, the verb *'essen'*, which means 'to eat' turns into a noun *'Essen'*, which means 'food' and this can also be experienced by other verbs. Here is an example of the derivation of the word *'lesen'*, which means 'to read', and it changes quoted by Gallmann. The word *'lesen'* changes to *lese, liest, las, lasest, läse, läsen, lies!, lesend, lesendes, lesenden, gelesen, gelesenes, Gelesenes, Gelesenen, Lesendes, Lesenden, Lesen, Lesens,* [4] and *Leser, Lesern, Lesers, lesbar*. Other verbs would experience these changes, such as in the example. To help identify the changes in derivational processes, Lemmatizer SpaCy can be utilized to analyze the changes in German vocabulary to determine its original/basic form and its inflection. .

## 3. NATURAL LANGUAGE PROCESSING (NLP)

The lemmatization process is carried out using the NLP method. Thus, the computer's understanding depends heavily on how well the setting of the morphology, syntax, semantics, phonetics, and grammar in the system which is called as a model language library. The better the system model language library provided in the computer, the better computer understanding of human language is, because the main task of NLP is to help the machines understand and respond to human language [5].

With the NLP method, the computer can read a text, hear and understand speeches, interpret, measure and classify sentiments, and determine essential sentence parts. In NLP, tokenization refers to the process of breaking text into small pieces called tokens (Kaushal et al., 2020). Besides, NLP is used to manage segmentation, tokenization, lemmatization, POS tagging, and NER [6]. Thus, in general, it can be stated that the task of NLP is to break the language into pieces of shorter sentence elements, then understand the relationships between the components, interrelate the details, and work together to create meaning [7] According to[8] in NLP, several terms need to be recognized, including token, tokenization, corpus, Part-of-Speech (POS)-Tag, and parse.

However, the larger the number of texts, the more difficult it is for the text to be disseminated to spread the knowledge contained in the text. However, NLP is considered to be effective and accurate in doing the process for the limited number of texts, just as humans do [9].

## 4. NATURAL LANGUAGE TOOLKIT (NLTK)

Python is software for a popular programming language. However, Python is not reliable enough to carry out more complex text analysis needs, such as lemmatization. This requires a sub-application called the Natural Language Toolkit and commonly abbreviated as NLTK. Lemmatization is the primary function in the NLP and NLTK software. Although they play a critical role, there are limited Lemmatizers for German [10]. Based on google search, at least four free Lemmatizers, namely GermaLemma, SpaCy, HanTa, and HanTa Hybrid. In this paper, Lemmatizer SpaCy is used for lemmatization. The use of SpaCy is based on

several considerations, including ease of installation and ease of operation, as well as the accuracy of the analysis results.

## 5. INSTALLING PYTHON

Python is a programming language software that is relatively easy for users to learn. It can run on operating systems Windows, Linux, and Macintosh. Based on the survey conducted, Python is a software programming language ranked five in the most widely used category in the whole world [11]. Python software can be downloaded via https://www.python.org/. Installing Python can be done like any other software. Python is open-source software, meaning that anyone can download and use Python freely [12], and it is currently becoming very popular among programmers. Besides, in recent years, Python called SpaCy can perform sentiment analysis in languages other than English because of its multilingual supports [13].

## 6. INSTALLING SPACY

SpaCy is an effective and efficient open-source NLP library dealing with NLP problems [14]. Following are the steps for installing SpaCy:

*a) Open a command prompt with Run as administrator.*

*b) Change directory to c: \>*

*c) Type: conda install -c conda-forge spacy or pip install -U spacy*

*d) Type: Python -m spacy download en*

The word *en* refers to English. Users can use other language library models, for example, German, France, Spanish, Portuguese, Italian, Dutch, Greek, and other languages. A list of languages that can be analyzed with Lemmatizer SpaCy can be seen at https://spacy.io/models/de, including Bahasa Indonesia. However, not all features for Bahasa Indonesia are available like other languages. Some of the missing features are the PoS-tagging, Named Entity Recognition (NER), and dependency parsing [15].

## 7. INSTALLING VISUAL STUDIO CODE

The VSC software can be downloaded on the https://code.visualstudio.com/download, and it is open-source software. This software is available in several OSs, such as Windows, Debian, Ubuntu, Red Hat, Fedora, SUSE, and macOS. To use VSC, users must download the installer first and install it on a computer device.

## 8. HOW TO RUN SPACY IN VISUAL STUDIO CODE

Lemmatizer SpaCy is used to determine the lemma form from a root word that has changed due to derivational processes. To minimize the complexity of the analysis procedure with Python, the author uses VSC software, which functions to run Python and the SpaCy Lemmatizer in one software, as shown in the following figure:

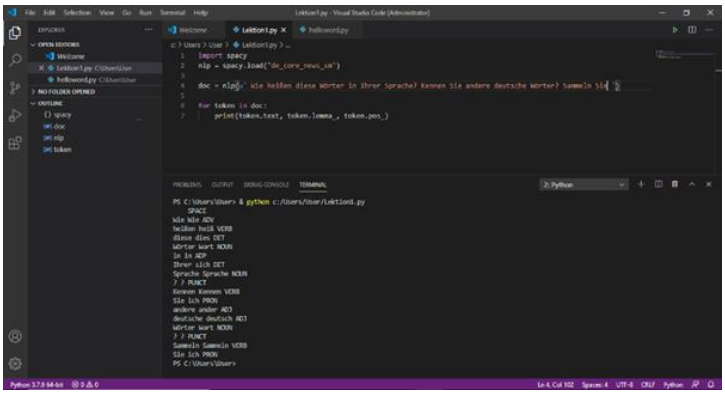

**Figure 1 SpaCy and Python collaboration in Visual Studio Code**

Assisted with the VSC, Lemmatizer SpaCy uses a programming language code that looks as follows

```
import spacy
nlp = spacy.load("de_core_news_sm")
doc = nlp("Gerade am Stadtrand hält Berlin historische Schätze bereit. Unsere heutige Entdeckungsreise zu verborgenen Perlen führt nach Blankenfelde.")
for token in doc:
    print(token.text, token.lemma_, token.pos_)
```

**Figure 2 Lematization process code**

The paragraph text entered in the column is analyzed based on the SpaCy language library model. The sentences in the paragraph are then parsed by word (tokenization), and the token, lemma, and PoS-tag are displayed. The examples of the results of how SpaCy Lemmatizer analyzes sentences in paragraphs can be seen in the following table:

**Table 1: Results of the lemmatization analysis by SpaCy****

| Token | Lemma | PoS-tag | Due |
|---|---|---|---|
| Gerade | Gerade | ADV | |
| am | am | ADP | |
| Stadtrand | Stadtrand | PROPN* | NOUN |
| hält | halten | VERB | |
| Berlin | Berlin | PROPN | |
| historische | historische* | ADJ | |
| Schätze | Schatz | NOUN | |
| bereit | bereiten | ADJ* | VERB |
| . | . | PUNCT | |
| Unsere | mein | DET | |
| heutige | heutige* | ADJ | heutig |
| Entdeckungsreise | Entdeckungsreise | NOUN | |
| zu | zu | ADP | |
| verborgenen | verborgen | ADJ | |
| Perlen | Perle | NOUN | |
| führt | führen | VERB | |
| nach | nach | ADP | |
| Blankenfelde | Blankenfelde | NOUN* | PROPN |
| . | . | PUNCT | |

* error analysis results

** results in Visual Studio Code are not tabular

Based on the lemmatization results above, Lemmatizer SpaCy can show the token, lemma, and PoS-tag form of a word in German, although there are errors in its analysis. In the table above, errors are marked with a sign (*).

Based on the results' analysis, SpaCy did not make an error in the PoS-tags of PUNCT, ADP, ADV because these words do not change the form, either inflection or derivational processes. Based on several experiments, SpaCy could make mistakes in the analysis of NOUN, PRON, ADJ, VERB, PART, and AUX, especially words that are inflection or derivation. Also, one of SpaCy's weaknesses is analyzing verbs that have the function as both full verbs and auxiliary verbs, for example, the verbs *haben*, (to have), *sein* (to be), and *werden* (to become).

## 9. CONCLUSIONS

SpaCy, in collaboration with Python and VSC, lemmatizes German texts through the analysis process at the word level. Based on the lemmatization results above, Lemmatizer SpaCy can show the form of token, lemma, and PoS-tag in German, although there are some errors in its analysis. This is motivated by several factors, including homographs, the grammar of a language, and other systems of grammatical rules. The inability of this analysis is one of the weaknesses of the available Lemmatizers.

## REFERENCES

[1] R. Ilson, (1988). Introduction. *International Journal of Lexicography*, *1*(1), 1-s-1. https://doi.org/10.1093/ijl/1.1.1-s

[2] Kementerian Kementerian Pendidikan dan Kebudayaan. (2019). *Petunjuk teknis penyusunan kamus Ekabahasa*. Pusat Pengembangan dan Pelindungan Bahasa dan Sastra Badan

Pengembangan Bahasa dan Perbukuan Kementerian Pendidikan dan Kebudayaan. http://badanbahasa.kemdikbud.go.id/lamanbahasa/sites/default/files/juknis/Juknis%20Penyusunan%20Kamus%20Ekabahasa.pdf

[3] I. S. P., Nation& S. Hunston, (2018). *Learning vocabulary in another language*. Cambridge Core; Cambridge University Press. https://doi.org/10.1017/CBO9781139858656

[4] P. Gallmann, (1991). Wort, lexem und lemma. In *Rechtschreibwörterbücher in der diskussion. geschichte – analyse – perspektiven. Augst, Gerhard / Schaeder, Burkhard (Hrsg.) (1991): Frankfurt am Main / Bern / New York / Paris:* (pp. 261-280.). Peter Lang. http://www.personal.uni-jena.de/~x1gape/Pub/Lemma_1991.pdf

[5] Y. Vasiliev, (2020). *Natural language processing with Python and SpaCy: A Practical Introduction*. No Starch Press.

[6] K., Weiying, D.N. Pham, Y. Eftekharypour,&A.J. Pheng (2019). Benchmarking NLP toolkits for enterprise application. In A.C. Nayak &A. Sharma (Eds.), PRICAI 2019 Trends in Artificial Intelligence (pp.289-294). Springer International Publishing. https://doi.org/10.1007/978-3-030-29894-4_24

[7] H. E.Rosyadi, , F Amrullah,., R. D., Marcus, & R. R. Affandi, (2020). Rancang bangun Chatbot informasi lowongan pekerjaan berbasis Whatsapp dengan metode NLP (Natural Language Processing). *Briliant: Jurnal Riset dan Konseptual*, *5*(3), 619–626. https://doi.org/10.28926/briliant.v5i3.487

[8] K. Fuadi, (2013). *Pengenalan NLP (Natural Language Processing) dengan Python*. Jogjakarta. https://kholidfu.github.io/assets/python_nltk_docs.pdf

[9] K. R. Chowdhary, (2020). Natural language processing. In K. R. Chowdhary (Ed.), *Fundamentals of artificial intelligence* (pp. 603–649). Springer India. https://doi.org/10.1007/978-81-322-3972-7_19

[10] C. Wartena, (2019). A probabilistic morphology model for German lemmatization. *KONVENS*, 11.

[11] I. B. Trisno, (2016). Belajar pemrograman sulit? Coba Python. In Y. Hari (Ed.), *Buku Ajar*. Ubahara Manajemen Press Surabaya. http://repository.widyakartika.ac.id/127

[12] A., Kedia& M Rasu,. (2020). *Hands-On Python natural language processing: Explore tools and techniques to analyze and process text with a view to building real-world NLP applications* (1st Edition). Packt Publishing.

[13] M. Sharma, (2020). Polarity detection in a cross-lingual sentiment analysis using spaCy. *8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, 490–496. https://doi.org/10.1109/ICRITO48877.2020.9197829

[14] Admin. (2019). *Instalasi dan Dasar spaCy – SkillPlus*. SkillPlus Free Indonesian Tutorial. https://skillplus.web.id/instalasi-dan-dasar-spacy/

[15] Bagas. (2018). *SpaCy bahasa Indonesia*. SpaCy Bahasa Indonesia. https://bagas.me/spacy-bahasa-indonesia.html