

TextBlob: A Text analysis tool for NLP tasks

Arya Brijith

IIPP Research Intern

Asia University

Taichung, Taiwan

(e-mail: arya.brijithk@gmail.com)

Abstract

This article is based on TextBlob, one of the most widely used tools in the Python programming language's "nltk" package. We will learn about how to install TextBlob, its fundamental characteristics, its use in sentiment analysis, its real-world application, and a lot more.

Keywords: TextBlob, sentiment analysis, NLP

Introduction

In the field of natural language processing and text analysis, TextBlob is your reliable partner.

A Python module called TextBlob, present in the 'nltk' library is intended to manage various activities connected to text. It can assist you with tasks like interpreting a customer review's mood, categorizing news items, and translating messages into several languages. Its simplicity is what makes TextBlob so appealing. Using it does not require you to be an expert coder or language scholar. A few lines of Python code may be used to tokenize text, identify the parts of speech used in a phrase, and even ascertain the emotion of a piece of text.

It goes beyond the fundamentals, though. With capabilities like language translation and recognition, TextBlob goes above and beyond. A text may be easily translated from one language to another by using this feature, which can recognize the language in which it is written. TextBlob is like a multipurpose Swiss Army knife in the field of text analysis, capable of handling a wide range of jobs. TextBlob is your dependable friend for delving into the

wonderful world of text, regardless of your background—you may be a data scientist, content curator, or just someone trying to learn the language of the web.

Now let us dive into the installation of TextBlob and various other information.

Installation

- **On Conda:**
\$ conda install -c conda-forge textblob
\$ python -m textblob.download_corpora
- **On Github source:**

You can also clone a public repo from GitHub. The code is as follows:

```
git clone https://github.com/sloria/TextBlob.git
python setup.py install # inside the cloned directory
```

For importing the TextBlob package from the library, simply type-

```
from textblob import TextBlob
```

We shall discuss an example code further in the article.

Features of TextBlob

TextBlob offers a range of tools and functions to facilitate the manipulation of textual data. Some of them are as follows:

- **POS (Parts of Speech) Tagging:** It is helpful for syntactic analysis since it can assign grammatical components of speech (such as nouns, verbs, and adjectives) to individual words in a phrase. It is helpful for syntactic analysis since it can assign grammatical components of speech (such as nouns, verbs, and adjectives) to individual words in a phrase.
- **Noun Phrase Extraction:** One linguistic feature in TextBlob is noun phrase extraction, which recognizes and extracts noun phrases automatically from text. Noun phrases are significant syntactic structures made up of nouns and the modifiers that go with them, including determiners and adjectives. They are essential for comprehending the main ideas and subjects that are covered in a book. This capability may be used for a variety of text analysis tasks, such as content interpretation, question answering, and information retrieval and summarization. TextBlob is an invaluable tool for anybody looking to identify and interact with essential concepts in textual material since it makes this process simpler and makes it easy for users to access extracted noun phrases.
- **Tokenization:** Tokenization is the act of dividing a text or document into smaller parts, usually words or sentences, to make text processing and language analysis easier. Words are typically referred to as tokens in NLP, and tokenization is a key process in transforming unstructured text into a format that is organized and suitable for computer analysis. Tokenization is the process of locating and classifying individual words, punctuation, and other language components inside a document. This serves as the basis for several NLP activities, including text categorization, sentiment analysis, and part-of-speech

tagging, in addition to making text easier to process and analyze. Furthermore, tokenization is language-dependent, which means that to guarantee correct findings in analyses tailored to a particular language, various languages may need distinct tokenization rules and algorithms.

- **Sentiment Analysis:** The sentiment analysis function of TextBlob is a useful tool for determining the emotional polarity or tone of a text. Text is given a polarity score, with 0 denoting a neutral attitude and ranging from -1 (very negative) to 1 (highly positive). The sentiment analysis function makes these judgments by utilizing a pre-trained sentiment classifier that combines pattern recognition and natural language processing methods. Later in the article, we shall discuss more about this.
- **Text Classification:** Text classification is made possible by TextBlob, which lets users group text documents into pre-established classes or categories. This is very helpful for jobs like sentiment analysis, subject classification, spam detection, and more. In TextBlob, text classification is the process of using labeled data—usually text documents with matching class labels—to train a text classifier. After training, the classifier uses the characteristics and patterns it has discovered to estimate the class of newly discovered, unlabeled text documents. The procedure is made easier by TextBlob's simple API, which even those with little background in machine learning and natural language processing can understand.

The utility of TextBlob in sentiment analysis

TextBlob is used to fetch the polarity and the subjectivity of the statements. The range of polarity is $[-1,1]$, where 1 denotes a positive attitude and -1 denotes a negative sentiment. Subjectivity measures how much of the

NGTIM

content is made up of factual data and personal views. Once we fetch the polarity, we classify the sentiment as positive(values>0), neutral(values=0), or negative(values<0).

Here is an example code, to perform sentiment analysis:

```
from textblob import TextBlob

# Create a sample text
text = "TextBlob ia a Text analysis tool for NLP tasks, it is amazing!!!"

blob = TextBlob(text)

# Perform sentiment analysis
polarity = blob.sentiment.polarity # Polarity measures sentiment (-1 to 1)
subjectivity = blob.sentiment.subjectivity # Subjectivity measures opinion (0 to 1)

# Determine sentiment based on polarity
if polarity > 0:
    sentiment = "positive"
elif polarity < 0:
    sentiment = "negative"
else:
    sentiment = "neutral"

# Print the results
print("Text:", text)
print("Polarity:", polarity)
print("Subjectivity:", subjectivity)
print("Sentiment:", sentiment)
```

Output:

```
Text: TextBlob ia a Text analysis tool for NLP tasks, it is amazing!!!
Polarity: 1.0
Subjectivity: 0.9
Sentiment: positive
```

Real-world applications

Some real-world TextBlob applications are as follows.

- **Sentiment analysis of customer feedback:** TextBlob is a tool used by companies to analyze customers' opinions, feedback, and behavior towards their products and services. Sentiment analytics make it easier to understand customer sentiment, identify areas for improvement, and enable data-driven decision-making to increase customer satisfaction
- **Social media monitoring:** Companies and brands target social media sites to discuss, discuss, and talk about their products and services. Sentiment analysis and trend tracking are enabled by TextBlob and can be used for customer interaction and brand name management.
- **Content structures:** TextBlob is used by article recommendation systems to analyze and categorize news articles, blog posts, and other types of content. This allows platforms to offer relevant content recommendations to

users based on their preferences and areas of interest.

- **Spam detection:** TextBlob is used by online forums and email providers to detect spam. TextBlob can help reduce unwanted and potentially dangerous text by analyzing the content of incoming communications and categorizing email content as genuine or spam
- **News Topic Classification:** TextBlob is used by news organizations and aggregators to categorize news stories in sports, politics, technology, and more. This allows the content to be organized and provides readers with information relevant to their interests.

This example demonstrates how TextBlob can be used to consume and analyze human-generated content in a variety of contexts, such as social media, marketing, content management, and recommendations.

Limitations

While TextBlob is useful and easy to navigate for text analysis, users should be aware of its limitations. First, it could not always give very accurate results, especially for specific tasks or domains. It is possible that pre-trained TextBlob instances are not optimized for specific words or domain-specific words. When manipulating task-specific features, this limitation may influence the accuracy of tasks such as text segmentation, sentiment analysis, or part-of-speech marking

Second, in complex applications, TextBlob's performance could not match sophisticated NLP libraries such as spaCy or deep learning-based models. Despite being well suited for applications, TextBlob may not work well with languages outside its functionality because its language support is limited to languages that are part of its models. Users who need full language support may need to look elsewhere. Despite these drawbacks, TextBlob is still a good choice for quick and easy text analysis projects, especially for people who are unfamiliar with natural language processing

(NLP) or need a quick and easy approach does deal with specific writing tasks

Conclusion

In conclusion, TextBlob is an intuitive and simple tool for text analysis and natural language processing. With its extensive features and user-friendly design, it is an ideal tool for novice and experienced NLP professionals alike. TextBlob makes processing text data easy, whether you're looking for sentiment analysis, language translation, or other real-world applications.

TextBlob's usefulness is evident in a variety of applications, including content curation, spam detection, consumer instincts, and helping businesses understand social media monitoring but it's important to understand its limitations, especially when working with sophisticated solutions to difficult languages or complex tasks that may require more. Despite these limitations, TextBlob is a reliable way to analyze text quickly and efficiently, making it an essential tool for anyone who wants to harness the power of human-powered text data to use it completely.

References

1. Verma, P. (2022). Sentiment Analysis Using Subjectivity and Sentiment Score. *The Next Normal: Building Agile, Sustainable, Tech-enabled Organizations (TNN-BASTO-2022)*, 55.
2. Loria, S. (2018). textblob Documentation. *Release 0.15*, 2(8), 269.
3. Gujjar, J. P., & Kumar, H. P. (2021). Sentiment analysis: Textblob for decision making. *Int. J. Sci. Res. Eng. Trends*, 7(2), 1097-1099.
4. Diyasa, I. G. S. M., Mandenni, N. M. I. M., Fachrurrozi, M. I., Pradika, S. I., Manab, K. R. N., & Sasmita, N. R. (2021, May). Twitter Sentiment Analysis as an Evaluation and Service Base On Python Textblob. In *IOP Conference Series: Materials Science and Engineering* (Vol. 1125, No. 1, p. 012034). IOP Publishing.
5. Kaur, C., & Sharma, A. (2020, October). Social issues sentiment analysis using python. In *2020 5th international conference on computing, communication and security (ICCCS)* (pp. 1-6). IEEE.
6. Elgendy, I. A., Zhang, W. Z., He, H., Gupta, B. B., El-Latif, A., & Ahmed, A. (2021). Joint computation offloading and task caching for multi-user and multi-task MEC systems: reinforcement learning-based algorithms. *Wireless Networks*, 27(3), 2023-2038.
7. Kumar, N., Poonia, V., Gupta, B. B., & Goyal, M. K. (2021). A novel framework for risk assessment and resilience of critical infrastructure towards climate change. *Technological Forecasting and Social Change*, 165, 120532.
8. Kaur, M., Singh, D., Kumar, V., Gupta, B. B., & Abd El-Latif, A. A. (2021). Secure and energy efficient-based E-health care framework for green internet of things. *IEEE Transactions on Green Communications and Networking*, 5(3), 1223-1231.
9. Hammad, M., Alkinani, M. H., Gupta, B. B., El-Latif, A., & Ahmed, A. (2021). Myocardial infarction detection based on deep neural network on imbalanced data. *Multimedia Systems*, 1-13.