Technische Beschreibung des NLP Workbench Python-Moduls

Anonym
ZHAW School of Engineering
Bachelor in Data Science

Abstract

Die NLP Workbench ist ein modulares, auf Python basierendes Framework, das für Ausbildungszwecke in einem Natural Language Processing-Kurs entwickelt wurde. Sie besteht aus drei Hauptkomponenten, wobei jedes Modul ist darauf ausgelegt, bestimmte Aufgaben in der Textanalyse-Pipeline auszuführen, die von der Vorverarbeitung einzelner Texte bis zur Analyse umfassender Textkorpora reichen.

1 Einführung

Das Verstehen und die Verarbeitung von Textdaten sind grundlegende Aspekte des NLP. Die NLP Workbench ist so aufgebaut, dass sie den Studierenden ein praktisches Werkzeug zur Erforschung dieser Prozesse bietet. Sie nutzt robuste Python-Bibliotheken, um wesentliche NLP-Aufgaben zu implementieren und erleichtert ein tieferes Verständnis von Textanalysemethoden.

2 Module Beschreibung

Das objektorientierte Design des Codes trennt die Funktionen effektiv in verschiedene Klassen auf. Die Klassen `TextPreprocessing` und `TextAnalyser` sind für die Analyse von Einzeltexten konzipiert und benötigen bei der Instanziierung keinen Korpus, was eine grössere Flexibilität und Modularität ermöglicht. Die Klasse `CorpusAnalyser` hingegen ist für die Analyse auf Korpusebene konzipiert und benötigt bei der Instanziierung den gesamten Korpus, um Vorverarbeitungen auf dem Korpus durchzuführen.

2.1 TextPreprocessing.py

Die Klasse `TextPreprocessing` kapselt die für die Textanalyse erforderlichen Vorverarbeitungsschritte. Sie verwendet Methoden zur Tokenisierung, Normalisierung und Bereinigung von Textdaten, um sicherzustellen, dass der Text in einer optimalen Form für die weitere Analyse vorliegt. Die Klasse abstrahiert diese Schritte, so dass Benutzer Texte vorverarbeiten können, ohne sich mit den zugrunde liegenden komplexen Verfahren auseinandersetzen zu müssen.

2.2 TextAnalyser.py

Die Klasse `TextAnalyser` wurde entwickelt, um detaillierte Analysen an einzelnen Texten durchzuführen. Sie enthält Methoden zum Zählen von Wörtern, zur Suche nach Teilstrings und für andere textspezifische Analysen. Durch die Kapselung dieser Funktionalitäten bietet die Klasse eine einfache Schnittstelle für Benutzer zur Durchführung von Textanalysen und fördert eine Umgebung, in der der Schwerpunkt auf den Analyseergebnissen und nicht auf den Implementierungsdetails liegt.

2.3 CorpusAnalyser.py

Die Klasse `CorpusAnalyser` erweitert die Analyse auf Sammlungen von Texten. Sie nutzt den `TfidfVectorizer` aus Scikit-learn, um eine TF-IDF-Matrix zu erstellen, eine entscheidende Komponente für das Verständnis der Bedeutung von Wörtern in verschiedenen Dokumenten. Die Methode `get_n_highest_ids()`, ermöglicht es dem Nutzer die relevantesten Dokumente für bestimmte Begriffe zu identifizieren. Darüber hinaus enthält sie Funktionalitäten zur Erstellung eines invertierten Index und zur Analyse grundlegender Statistiken auf Korpusebene.

3 Conclusion

Die NLP Workbench bietet durch ihren modularen Aufbau eine umfassende Suite von Werkzeugen für die Text- und Korpusanalyse. Jedes Modul ist darauf ausgelegt, bestimmte Funktionen im Textanalyseprozess zu erfüllen, und bietet dem Benutzer die Möglichkeit, Texte vorzuverarbeiten und detaillierte Analysen durchzuführen.