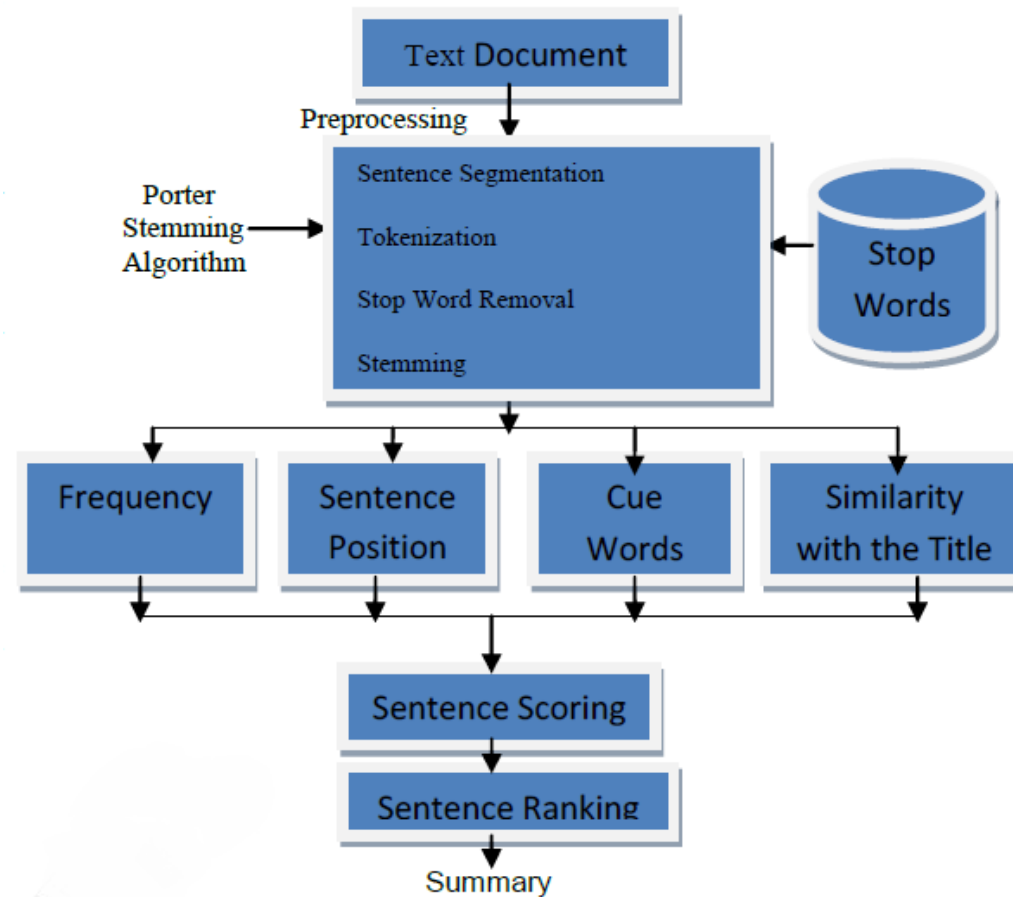


Text Summarizer in Java

Beispiele

Generelles Vorgehen



1. Beispiel (minimalistisch)

<https://github.com/karimo94/Text-Summarizer/blob/master/Summarizer.java>

- 160 Codezeile (inkl. Leerzeilen, Kommentare)



1. Beispiel (minimalistisch)

<https://github.com/karimo94/Text-Summarizer/blob/master/Summarizer.java>

- 160 Codezeile (inkl. Leerzeilen, Kommentare)
- 1 Klasse, 6 Methoden (5 private, 1 public)
- `Map<String, Integer> getWordCounts(String text)`
- `Map<String,Integer> filterStopWords(Map<String, Integer> d)`
- `List<String> sortByFreqThenDropFreq(Map<String,Integer> wordFrequencies)`
- `String[] getSentences(String text)`
- `String search(String[] sentences, String word)`
- `String Summarize(String text, int maxSummarySize)`

1. Beispiel (minimalistisch)

<https://github.com/karimo94/Text-Summarizer/blob/master/Summarizer.java>

- 160 Codezeile (inkl. Leerzeilen, Kommentare)
- 1 Klasse, 6 Methoden (5 private, 1 public)

Vorteile:

- Wenig Code
- Leicht zu verstehen

Nachteile:

- nur eine Sprache
- nicht konfigurierbar
- keine zusätzlichen Ausgaben (z.B. Statistik)

- `Map<String, Integer> getWordCounts(String text)`
- `Map<String,Integer> filterStopWords(Map<String, Integer> d)`
- `List<String>`
`sortByFreqThenDropFreq(Map<String,Integer>`
`wordFrequencies)`
- `String[] getSentences(String text)`
- `String search(String[] sentences, String word)`
- `String Summarize(String text, int maxSummarySize)`

2. Beispiel

<https://github.com/ajhalthor/text-summarizer>

- Klassenbasierter Ansatz
- Vorgehen:
 1. Kontext in Absätze teilen
 2. Absätze in Sätze teilen
 3. Bestimme Anzahl gemeinsame Worte zwei beliebigen Sätzen (gewichtet mit mittlerer Länge der beiden Sätze)
 4. Werte in Matrix zwischenspeichern
 5. Finaler Punktestand in Dictionary speichern (Satz, Punkte)
 6. Sätze mit höchstem Punktestand extrahieren
 7. Auswahl chronologisch ordnen

2. Beispiel

<https://github.com/ajhalthor/text-summarizer>

Klassen:

- Sentence

```
class Sentence {  
    int paragraphNumber;  
    int number;  
    int stringLength; //Dont need this  
    double score;  
    int noOfWords;  
    String value;  
    Sentence (int number, String value, int stringLength,  
              int paragraphNumber) {  
        this.number = number;  
        this.value = new String(value);  
        this.stringLength = stringLength;  
        noOfWords = value.split("\\s+").length;  
        score = 0.0;  
        this.paragraphNumber = paragraphNumber;  
    }  
}
```

2. Beispiel

<https://github.com/ajhalthor/text-summarizer>

Klassen:

- Sentence

Speichert:

- Position im Text
- Zugehöriger Absatz
- Satzlänge
- Anzahl Worte
- Punkte
- Satzstring

2. Beispiel

<https://github.com/ajhalthor/text-summarizer>

Klassen:

- Sentence
- Paragraph

```
import java.util.ArrayList;

class Paragraph{
    int number;
    ArrayList<Sentence> sentences;

    Paragraph(int number){
        this.number = number;
        sentences = new ArrayList<Sentence>();
    }
}
```

2. Beispiel

<https://github.com/ajhalthor/text-summarizer>

Klassen:

- Sentence
- Paragraph

Speichert

- Absatznummer
- Liste von Sätzen

2. Beispiel

<https://github.com/ajhalthor/text-summarizer>

Klassen:

- Sentence
- Paragraph
- SentenceComparator

```
import java.util.Comparator;

class SentenceComparator implements Comparator<Sentence> {
    @Override
    public int compare(Sentence obj1, Sentence obj2) {
        if (obj1.score > obj2.score) {
            return -1;
        } else if (obj1.score < obj2.score){
            return 1;
        } else {
            return 0;
        }
    }
}
```

2. Beispiel

<https://github.com/ajhalthor/text-summarizer>

Klassen:

- Sentence
- Paragraph
- SentenceComparator
- Vergleicht Sätze anhand deren Punktstand
- Dient zur Auswahl von Sätzen für die Zusammenfassung

2. Beispiel

<https://github.com/ajhalthor/text-summarizer>

Klassen:

- Sentence
- Paragraph
- SentenceComparator
- SentenceComparatorForSummary

```
import java.util.Comparator;

class SentenceComparatorForSummary implements
    Comparator<Sentence>{
    @Override
    public int compare(Sentence obj1, Sentence obj2) {
        if (obj1.number > obj2.number) {
            return 1;
        } else if (obj1.number < obj2.number) {
            return -1;
        } else {
            return 0;
        }
    }
}
```

2. Beispiel

<https://github.com/ajhalthor/text-summarizer>

Klassen:

- Sentence
- Paragraph
- SentenceComparator
- SentenceComparatorForSummary
- Vergleicht Sätze anhand deren Position
- Dient zum chronologischen ordnen

2. Beispiel

<https://github.com/ajhalthor/text-summarizer>

Klassen:

- Sentence
- Paragraph
- SentenceComparator
- SentenceComparatorForSummary
- SummaryTool
- Instanziierung/Initialisierung
- Lesen des Kontexts
- Primitiver Tokenizer
- Berechnung der Punkte
- Erstellen und Ausgabe der Zusammenfassung

2. Beispiel

<https://github.com/ajhalthor/text-summarizer>

Vorteile:

- Berücksichtigt Absatzlänge
- Sprachunabhängig
- Zusätzliche Ausgaben

Nachteile:

- Primitiver Tokenizer
- Kein Stemming
- Keine Stop Wörter