1. Токенизация с использованием Scanner

import java.util.Scanner;

import java.util.ArrayList;

import java.util.List;

public class TokenizationWithScanner {

public static void main(String[] args) {

Scanner scanner = new Scanner("Let's pause, and then reflect.");

List<String> list = new ArrayList<>();

scanner.useDelimiter("[ ,.]");

while (scanner.hasNext()) {

String token = scanner.next();

list.add(token);

}

for (String token : list) {

System.out.println(token);

}

scanner.close();

}

}

2. Токенизация с использованием split

public class TokenizationWithSplit {

public static void main(String[] args) {

String text = "Mr. Smith went to 123 Washington avenue.";

String tokens[] = text.split("\\s+");

for (String token : tokens) {

System.out.println(token);

}

text = "Let's pause, and then reflect.";

tokens = text.split("[ ,.]");

for (String token : tokens) {

System.out.println(token);

}

}

}

3. Токенизация с использованием BreakIterator

import java.text.BreakIterator;

public class TokenizationWithBreakIterator {

public static void main(String[] args) {

BreakIterator wordIterator = BreakIterator.getWordInstance();

String text = "Let's pause, and then reflect.";

wordIterator.setText(text);

int boundary = wordIterator.first();

while (boundary != BreakIterator.DONE) {

int begin = boundary;

boundary = wordIterator.next();

int end = boundary;

if (end == BreakIterator.DONE) break;

System.out.println(boundary + " [" + text.substring(begin, end) + "]");

}

}

}

4. Токенизация с использованием StreamTokenizer

import java.io.StringReader;

import java.io.IOException;

import java.io.StreamTokenizer;

public class TokenizationWithStreamTokenizer {

public static void main(String[] args) {

try {

StreamTokenizer tokenizer = new StreamTokenizer(new StringReader("Let's pause, and then reflect."));

boolean isEOF = false;

while (!isEOF) {

int token = tokenizer.nextToken();

switch (token) {

case StreamTokenizer.TT\_EOF:

isEOF = true;

break;

case StreamTokenizer.TT\_WORD:

System.out.println(tokenizer.sval);

break;

case StreamTokenizer.TT\_NUMBER:

System.out.println(tokenizer.nval);

break;

default:

System.out.println((char) token);

}

}

} catch (IOException ex) {

ex.printStackTrace();

}

}

}

5. Токенизация с использованием StringTokenizer

import java.util.StringTokenizer;

public class TokenizationWithStringTokenizer {

public static void main(String[] args) {

StringTokenizer st = new StringTokenizer("Let's pause, and then reflect.");

while (st.hasMoreTokens()) {

System.out.println(st.nextToken());

}

}

}

6. Токенизация с использованием OpenNLP

import opennlp.tools.tokenize.SimpleTokenizer;

import opennlp.tools.tokenize.TokenizerME;

import opennlp.tools.tokenize.TokenizerModel;

import java.io.FileInputStream;

import java.io.InputStream;

public class TokenizationWithOpenNLP {

private static String paragraph = "Let's pause, and then reflect.";

public static void main(String[] args) {

usingTheSimpleTokenizerClass();

// usingTheTokenizerMEClass(); // Uncomment to test TokenizerME

}

private static void usingTheSimpleTokenizerClass() {

System.out.println("--- SimpleTokenizer");

SimpleTokenizer simpleTokenizer = SimpleTokenizer.INSTANCE;

String tokens[] = simpleTokenizer.tokenize(paragraph);

for (String token : tokens) {

System.out.println(token);

}

}

// Example method for TokenizerME (to be uncommented)

private static void usingTheTokenizerMEClass() {

try {

InputStream modelIn = new FileInputStream("path/to/your/model.bin");

TokenizerModel model = new TokenizerModel(modelIn);

TokenizerME tokenizer = new TokenizerME(model);

String tokens[] = tokenizer.tokenize(paragraph);

for (String token : tokens) {

System.out.println(token);

}

} catch (Exception ex) {

ex.printStackTrace();

}

}

}

7. Токенизация с использованием Stanford NLP

import edu.stanford.nlp.pipeline.\*;

import edu.stanford.nlp.ling.\*;

import java.io.StringReader;

import java.util.Properties;

public class TokenizationWithStanford {

private static String paragraph = "Let's pause, and then reflect.";

public static void main(String[] args) {

usingTheStanfordTokenizer();

}

private static void usingTheStanfordTokenizer() {

Properties props = new Properties();

props.put("annotators", "tokenize");

StanfordCoreNLP pipeline = new StanfordCoreNLP(props);

Annotation document = new Annotation(paragraph);

pipeline.annotate(document);

for (CoreMap sentence : document.get(SentencesAnnotation.class)) {

for (CoreLabel token : sentence.get(TokensAnnotation.class)) {

System.out.println(token.word());

}

}

}

}

8. Примеры стемминга

import com.aliasi.tokenizer.Tokenization;

import com.aliasi.tokenizer.TokenizerFactory;

import com.aliasi.tokenizer.PorterStemmerTokenizerFactory;

import com.aliasi.tokenizer.IndoEuropeanTokenizerFactory;

public class StemmingExample {

public static void main(String[] args) {

usingTheLingPipeStemmer();

}

private static void usingTheLingPipeStemmer() {

String words[] = {"bank", "banking", "banks", "banker", "banked", "bankart"};

TokenizerFactory tokenizerFactory = IndoEuropeanTokenizerFactory.INSTANCE;

TokenizerFactory porterFactory = new PorterStemmerTokenizerFactory(tokenizerFactory);

for (String word : words) {

Tokenization tokenizer = new Tokenization(word, porterFactory);

String[] stems = tokenizer.tokens();

System.out.print("Word: " + word);

for (String stem : stems) {

System.out.println(" Stem: " + stem);

}

}

}

}

9. Примеры лемматизации

import edu.stanford.nlp.pipeline.\*;

import edu.stanford.nlp.ling.\*;

import java.util.Properties;

public class LemmatizationExample {

public static void main(String[] args) {

usingTheStanfordLemmatizer();

}

private static void usingTheStanfordLemmatizer() {

String paragraph = "Similar to stemming is Lemmatization. This is the process of finding its lemma.";

Properties props = new Properties();

props.put("annotators", "tokenize, ssplit, pos, lemma");

StanfordCoreNLP pipeline = new StanfordCoreNLP(props);

Annotation document = new Annotation(paragraph);

pipeline.annotate(document);

for (CoreMap sentence : document.get(SentencesAnnotation.class)) {

for (CoreLabel word : sentence.get(TokensAnnotation.class)) {

System.out.println(word.lemma());

}

}

}

}

10. Примеры работы со стоп-словами

import com.aliasi.tokenizer.TokenizerFactory;

import com.aliasi.tokenizer.IndoEuropeanTokenizerFactory;

import com.aliasi.tokenizer.EnglishStopTokenizerFactory;

import com.aliasi.tokenizer.Tokenizer;

public class StopWordsExample {

public static void main(String[] args) {

usingLingpipeStopWord();

}

private static void usingLingpipeStopWord() {

String paragraph = "A simple approach is to create a class to hold and remove stopwords.";

TokenizerFactory factory = IndoEuropeanTokenizerFactory.INSTANCE;

factory = new EnglishStopTokenizerFactory(factory);

Tokenizer tokenizer = factory.tokenizer(paragraph.toCharArray(), 0, paragraph.length());

while (tokenizer.hasNext()) {

System.out.println(tokenizer.next());

}

}

}

11. Примеры нормализации

public class NormalizationExample {

public static void main(String[] args) {

usingLingPipeForNormalization();

}

private static void usingLingPipeForNormalization() {

String paragraph = "A simple approach is to create a class to hold and remove stopwords.";

String result = paragraph.toLowerCase();

System.out.println(result + "\nЗавершено ЛингПайп--------");

}

}