

How to setup openNLP Java Project in Eclipse

How to setup openNLP Java Project

In this openNLP tutorial, we shall see **how to setup openNLP java project to use openNLP API** with Eclipse (the process should be same, to other IDEs as well).

Following are the steps to be followed :

1. Create a Java Project in the Eclipse. (Open Eclipse -> File(in Menu) -> New -> Project -> Java -> Java Project)
2. Provide a project name (Ex : OpenNLPJavaTutorial) and click on "Finish".
3. **Download jar files of openNLP** from [<http://redrockdigimark.com/apachemirror/opennlp/>] At the time of writing this tutorial, opennlp-1.7.1 is the latest, and the list looks like in the below picture

Index of /apachemirror/opennlp

Name	Last modified	Size	Description
Parent Directory		-	
opennlp-1.5.3/	15-Oct-2015 02:58	-	
opennlp-1.6.0/	15-Oct-2015 02:58	-	
opennlp-1.7.0/	01-Jan-2017 05:40	-	
opennlp-1.7.1/	24-Jan-2017 04:49	-	

opennlp version links

Click on opennlp-1.7.1/ . We need bin package, because that could have the library (.jar) files.

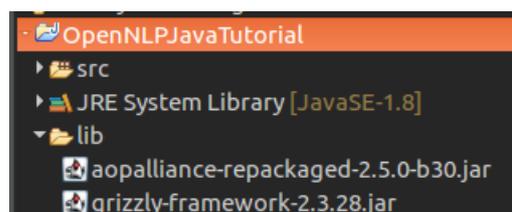
Index of /apachemirror/opennlp/opennlp-1.7.1

Name	Last modified	Size	Description
Parent Directory		-	
apache-opennlp-1.7.1-bin.tar.gz	24-Jan-2017 04:49	10M	
apache-opennlp-1.7.1-bin.zip	24-Jan-2017 04:49	13M	
apache-opennlp-1.7.1-src.tar.gz	24-Jan-2017 04:49	2.2M	
apache-opennlp-1.7.1-src.zip	24-Jan-2017 04:49	3.0M	

openNLP bin package

Click on apache-opennlp-1.7.1-bin.zip to download.

4. Once the zip file is downloaded, extract the contents, **copy the lib folder and paste in the project** as shown in the below picture.

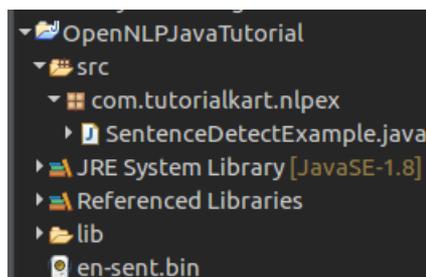


opennlp-java-project-lib folder

Lib folder should contain the list of below jar files: aopalliance-repackaged-2.5.0-b30.jar, grizzly-framework-

2.3.28.jargrizzly-http-2.3.28.jargrizzly-http-server-2.3.28.jarhk2-api-2.5.0-b30.jarhk2-locator-2.5.0-b30.jarhk2-utils-2.5.0-b30.jarhppc-0.7.1.jarjackson-annotations-2.8.4.jarjackson-core-2.8.4.jarjackson-databind-2.8.4.jarjackson-jaxrs-base-2.8.4.jarjackson-jaxrs-json-provider-2.8.4.jarjackson-module-jaxb-annotations-2.8.4.jarjavassist-3.20.0-GA.jarjavax.annotation-api-1.2.jarjavax.inject-2.5.0-b30.jarjavax.ws.rs-api-2.0.1.jarjcommander-1.48.jarjersey-client-2.25.jarjersey-common-2.25.jarjersey-container-grizzly2-http-2.25.jarjersey-entity-filtering-2.25.jarjersey-guava-2.25.jarjersey-media-jaxb-2.25.jarjersey-media-json-jackson-2.25.jarjersey-server-2.25.jarmorfologik-fsa-2.1.0.jarmorfologik-fsa-builders-2.1.0.jarmorfologik-stemming-2.1.0.jarmorfologik-tools-2.1.0.jaropennlp-brat-annotator-1.7.1.jaropennlp-morfologik-addon-1.7.1.jaropennlp-tools-1.7.1.jaropennlp-uima-1.7.1.jarosgi-resource-locator-1.0.1.jarvalidation-api-1.1.0.Final.jar

5. **Add these jars to the build path** (Project -> Properties -> Java Build Path -> Libraries -> Add Jars -> Select all the jars in lib folder -> Click "Apply" -> Click "OK")
6. Apache has already trained some models for different problems in Natural Language Processing, with training data, and these models are available at [<http://opennlp.sourceforge.net/models-1.5/>]. In the subsequent tutorials, we would refer to model files, which are available at this location. Do bookmark the link for a quick access.
7. We are ready with the openNLP Java Project Setup. Lets try Sentence detection using SentenceDetectExample.java.
8. Download "en-sent.bin" model file and place in the project. The final project structure should match with the structure shown in the below picture



opennlp java project structure

Example : We shall try out the example, SentenceDetectExample.java to check if the setup is good

SentenceDetectExample.java

```
import  
java.io.FileInputStream
```

```

import java.io.FileInputStream;
import java.io.IOException;
import java.io.InputStream;

import com.fasterxml.jackson.databind.exc.InvalidFormatException;

import opennlp.tools.sentdetect.SentenceDetectorME;
import opennlp.tools.sentdetect.SentenceModel;
/**
 * @author tutorialkart
 */
public class SentenceDetectExample {

    public static void main(String[] args) {
        try {
            new SentenceDetectExample().sentenceDetect();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    public void sentenceDetect() throws InvalidFormatException, IOException {
        String paragraph = "Apache openNLP supports the most common NLP tasks, such as tokenization, sentence
segmentation, part-of-speech tagging, named entity extraction, chunking, parsing, and coreference resolution. These tasks
are usually required to build more advanced text processing services. OpenNLP also includes maximum entropy and
perceptron based machine learning.";

        // refer to model file "en-sent.bin", available at link http://opennlp.sourceforge.net/models-1.5/
        InputStream is = new FileInputStream("en-sent.bin");
        SentenceModel model = new SentenceModel(is);

        // load the model
        SentenceDetectorME sdetector = new SentenceDetectorME(model);

        // detect sentences in the paragraph
        String sentences[] = sdetector.sentDetect(paragraph);

        // print the sentences detected, to console
        for(int i=0;i<sentences.length;i++){
            System.out.println(sentences[i]);
        }
        is.close();
    }
}

```

When SentenceDetectExample.java is run, the console output is:

Program Output

Apache openNLP
supports the most

Apache openNLP supports the most common NLP tasks, such as tokenization, sentence segmentation, part-of-speech tagging, named entity extraction, chunking, parsing, and coreference resolution. These tasks are usually required to build more advanced text processing services. OpenNLP also includes maximum entropy and perceptron based machine learning.

We are successfully done with the setup of openNLP Java Project in Eclipse.

Conclusion :

In this openNLP tutorial, we have seen the setup of openNLP Java Project in Eclipse. In our next openNLP tutorials, we shall see :

- [Named Entity Extraction](#)
- [Sentence Detection](#)

Learn OpenNLP

‣ [OpenNLP Tutorial](#)

‣ [Setup Java Project with OpenNLP in Eclipse](#)

‣ [OpenNLP Models](#)

Detection / Extraction using Java API

‣ [Tokenizer Example](#)

‣ [Sentence Detection Example](#)

‣ [Parts-Of-Speech Tagger Example](#)

‣ [Chunker Example](#)

‣ [Lemmatizer Example](#)

‣ [Named Entity Extraction Example](#)

Training using Java API

‣ [Sentence Detection Model Training](#)

‣ [Name Entity Finder Model Training](#)

‣ [Document Categorizer Training - Maximum Entropy](#)

‣ [Document Categorizer Training - Naive Bayes](#)

‣ [Document Categorizer with N-gram features used](#)

‣ [Language Detector Training Example](#)

Command Line Tools

‣ [Setup and start using Command Line Tools](#)

Useful Resources

‣ [How to Learn Programming](#)