How to use command line tools in Apache OpenNLP

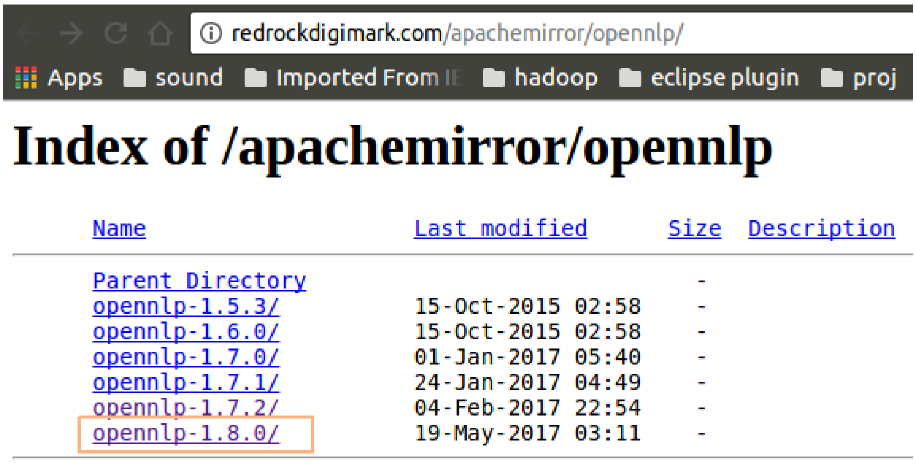
Command line tools in Apache OpenNLP

Command line tools in Apache OpenNLP – In this OpenNLP tutorial, we shall learn how to use command line tools that Apache OpenNLP provides to do natural language processing tasks like Named Entity Recognition (NER), Parts Of Speech tagging, Chunking, Sentence Detection, Document Classification or Categorization, Tokenization etc.

Following are the steps to setup command line tools in Apache OpenNLP :

- ### Step 1 : Download Apache OpenNLP.

  Click on the latest build of Apache OpenNLP from [http://redrockdigimark.com/apachemirror/opennlp/]

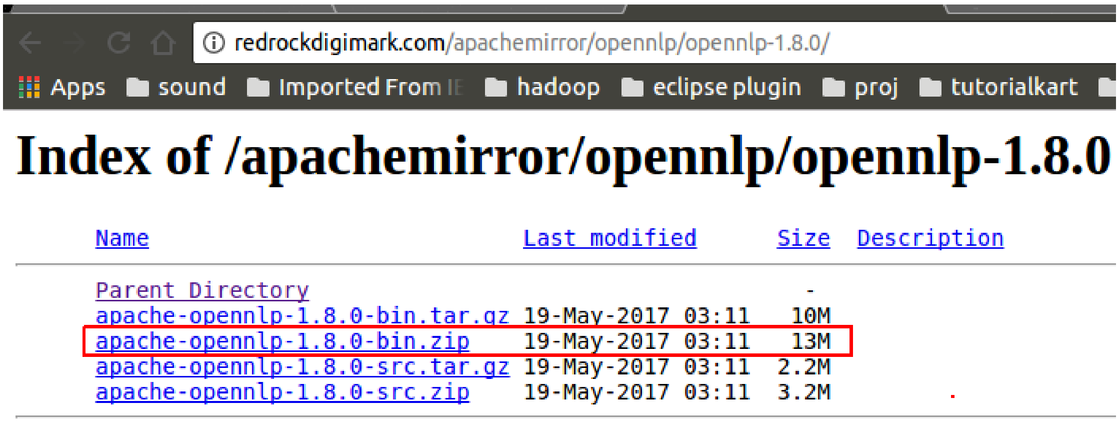


Index of /apachemirror/opennlp

```
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-1.7.2/       04-Feb-2017 22:54   -
opennlp-1.8.0/       19-May-2017 03:11   -
```

OpenNLP Mirror for Download

Click on the bin package (zip). We are not going to build it from source, we are just going to use the pre-built version

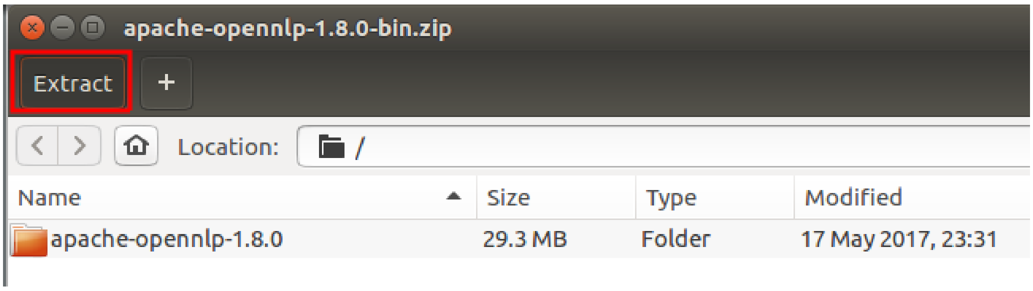


Index of /apachemirror/opennlp/opennlp-1.8.0

```
Name                            Last modified      Size  Description
Parent Directory                                     -
apache-opennlp-1.8.0-bin.tar.gz 19-May-2017 03:11   10M
apache-opennlp-1.8.0-bin.zip    19-May-2017 03:11   13M
apache-opennlp-1.8.0-src.tar.gz 19-May-2017 03:11  2.2M
apache-opennlp-1.8.0-src.zip    19-May-2017 03:11  3.2M
```

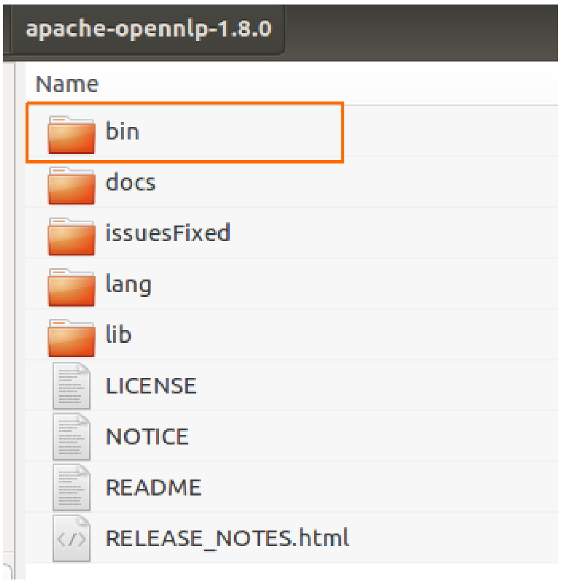
OpenNLP Built Package

- ### Step 2 : Unzip the package and navigate into bin folder.



Name	Size	Type	Modified
apache-opennlp-1.8.0	29.3 MB	Folder	17 May 2017, 23:31

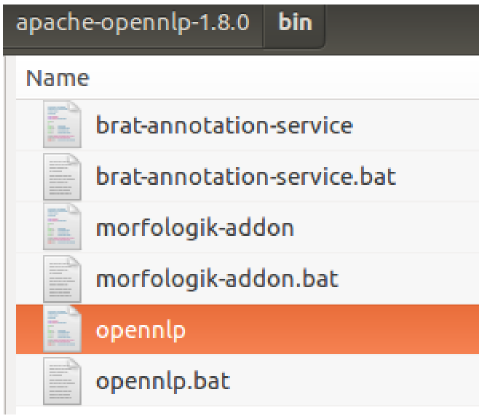
Extract contents from OpenNLP zip



Name

- bin
- docs
- issuesFixed
- lang
- lib
- LICENSE
- NOTICE
- README
- RELEASE_NOTES.html

OpenNLP bin



Name

- brat-annotation-service
- brat-annotation-service.bat
- morfologik-addon
- morfologik-addon.bat
- opennlp
- opennlp.bat

OpenNLP shell/batch file

For Ubuntu : Open the terminal and run the command

```
./opennlp
```

For Windows : Open the command prompt and give the command opennlp.bat

```
opennlp.bat
```

The following Usage of OpenNLP should be echoed on to the terminal or prompt :

Usage of OpenNLP

```
arjun@arjun-
```

```
arjun@arjun-VPCEH26EN:~/apache-opennlp-1.8.0/bin$ ./opennlp
OpenNLP 1.8.0. Usage: opennlp TOOL
where TOOL is one of:
  Doccat                         learned document categorizer
  DoccatTrainer                  trainer for the learnable document categorizer
  DoccatEvaluator                Measures the performance of the Doccat model with the reference data
  DoccatCrossValidator K-fold cross validator for the learnable Document Categorizer
  DoccatConverter                converts leipzig data format to native OpenNLP format
  DictionaryBuilder              builds a new dictionary
  SimpleTokenizer                character class tokenizer
  TokenizerME                    learnable tokenizer
  TokenizerTrainer               trainer for the learnable tokenizer
  TokenizerMEEvaluator           evaluator for the learnable tokenizer
  TokenizerCrossValidator K-fold cross validator for the learnable tokenizer
  TokenizerConverter             converts foreign data formats (ad,pos,conllx,namefinder,parse) to native OpenNLP format
  DictionaryDetokenizer
  SentenceDetector               learnable sentence detector
  SentenceDetectorTrainer        trainer for the learnable sentence detector
  SentenceDetectorEvaluator      evaluator for the learnable sentence detector
  SentenceDetectorCrossValidator K-fold cross validator for the learnable sentence detector
  SentenceDetectorConverter      converts foreign data formats (ad,pos,conllx,namefinder,parse,moses,letsmt) to native OpenNLP
format
  TokenNameFinder                learnable name finder
  TokenNameFinderTrainer         trainer for the learnable name finder
  TokenNameFinderEvaluator       Measures the performance of the NameFinder model with the reference data
  TokenNameFinderCrossValidator K-fold cross validator for the learnable Name Finder
  TokenNameFinderConverter       converts foreign data
formats (evalita,ad,conll03,bionlp2004,conll02,muc6,ontonotes,brat) to native OpenNLP format
  CensusDictionaryCreator        Converts 1990 US Census names into a dictionary
  POSTagger                      learnable part of speech tagger
  POSTaggerTrainer               trains a model for the part-of-speech tagger
  POSTaggerEvaluator             Measures the performance of the POS tagger model with the reference data
  POSTaggerCrossValidator K-fold cross validator for the learnable POS tagger
  POSTaggerConverter             converts foreign data formats (ad,conllx,parse,ontonotes,conllu) to native OpenNLP format
  LemmatizerME                   learnable lemmatizer
  LemmatizerTrainerME            trainer for the learnable lemmatizer
  LemmatizerEvaluator            Measures the performance of the Lemmatizer model with the reference data
  ChunkerME                      learnable chunker
  ChunkerTrainerME               trainer for the learnable chunker
  ChunkerEvaluator               Measures the performance of the Chunker model with the reference data
  ChunkerCrossValidator K-fold cross validator for the chunker
  ChunkerConverter               converts ad data format to native OpenNLP format
  Parser                         performs full syntactic parsing
  ParserTrainer                  trains the learnable parser
  ParserEvaluator                Measures the performance of the Parser model with the reference data
  ParserConverter                converts foreign data formats (ontonotes,frenchtreebank) to native OpenNLP format
  BuildModelUpdater              trains and updates the build model in a parser model
  CheckModelUpdater             trains and updates the check model in a parser model
  TaggerModelReplacer           replaces the tagger model in a parser model
  EntityLinker              links an entity to an external data set
  NGramLanguageModel              gives the probability and most probable next token(s) of a sequence of tokens in a language model
All tools print help when invoked with help parameter
Example: opennlp SimpleTokenizer help
arjun@arjun-VPCEH26EN:~/apache-opennlp-1.8.0/bin$
```

- ## Step 3 : Run opennlp command for help on any of the modules it presented in the above step

Help regarding any of the available task could be checked out using the Example mentioned in the response to opennlp command

```
$ ./opennlp SimpleTokenizer help
```

The response to the above command is shown below :

```
arjun@arjun-
VPCEH26EN:~/apache
```

```
arjun@arjun-VPCEH26EN:~/apache-opennlp-1.8.0/bin$ ./opennlp SimpleTokenizer help
Usage: opennlp SimpleTokenizer < sentences
```

- ## Step 4 : Lets try to actually use SimpleTokenizer

Create a text file, “sentences.txt” in the bin folder with sentences in it like below:

```
I am Joey. And I don’t share food. Welcome to friends.
```

Run the command

```
./opennlp SimpleTokenizer < sentences.txt
```

The following output of SimpleTokenizer on sentences.txt is echoed to the terminal or prompt

SimpleTokenizer Command Prompt Example

```
arjun@arjun-
VPCEH26EN:~/apache
```

```
arjun@arjun-VPCEH26EN:~/apache-opennlp-1.8.0/bin$ ./opennlp SimpleTokenizer < sentences.txt
I am Joey .
And I don ' t share food .
Welcome to friends .



Average: 750.0 sent/s
Total: 3 sent
Runtime: 0.004s
Execution time: 0.033 seconds
arjun@arjun-VPCEH26EN:~/apache-opennlp-1.8.0/bin$
```

SimpleTokenizer has found the tokens in the sentences and echoed on to the terminal. It also reported that there are three sentences in the file, “sentences.txt”.

Conclusion :

We have successfully learned how to setup and use Command Line Tools in Apache OpenNLP. In our furthur tutorials, we shall see how to do other Natural Language Processing tasks using Apache’s OpenNLP Command Line Tools.

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