OpenNLP – A Tutorial

http://opennlp.sourceforge.net/

Brett Walenz

University of Nebraska at Omaha

Background

- OpenNLP is a set of java-based Natural Language Processing tools using the Maximum Entropy mechanism, a statistical learning approach.
- OpenNLP can be used for:
  - Sentence Detection
  - Tokenization
  - Named-Entity Detection
  - Sentence Parsing
  - Coreference
  - Document Classification

Loading a Model

- All OpenNLP functions require a training model. Default models can be found at:
  - http://opennlp.sourceforge.net/models/english/
- Default models exist for all major tasks.
- General steps for using OpenNLP:
  - Identify the task and model
  - Train and build a GISModel file, if does not exist.
  - Load the model
  - Feed data to task

Loading a Model

- Named-Entity Task

```
GISModel model = new PooledGISModelReader(new
 File(modelFile)).getModel();
   NameFinderME nameFinder = new
NameFinderME(model);
```

`GISModel` is the trained model interface used by OpenNLP.
`[Application]ME` is the standard notation used for the various tasks' interfaces:

`NameFinderME`

`DocumentCategorizerME`

`TokenizerME`

Document Classification Example

- 1) Train models on existing documents

```
Collection<DocumentSample> samples = getSamples(directory)
builder = new DataStreamBuilder();
builder.add(samples)
dc = new DocumentCategorizerEventStream(builder)
GISModel model = DocumentCategorizerME.train(dc)
PlainTextGISModelWriter writer = new PlainTextGISModelWriter(model, file);
writer.persist(); //our model is now saved to disk
```

Document Classification Example

- 2) Use on incoming text
  - Load model

```
PlainTextGISModelReader reader = new PlainTextGISModelReader(file)
GISModel model = reader.getModel();
DocumentCategorizerME dc = new DocumentCategorizerME(model)
```

- 3) Categorize results

```
DocumentSample newData;
Double vals[] = categorizer.categorize(newData.getText())
//vals is an array of scores for every classification
```

Document Classification Example DocumentCategorizerME API

Method Summary	
double[]	categorize(java.lang.String documentText)
double[]	categorize(java.lang.String[] text) Categorizes the given text.
java.lang.String	getAllResults(double[] results)
java.lang.String	getBestCategory(double[] outcome)
java.lang.String	getCategory(int index)
int	getIndex(java.lang.String category)
int	getNumberOfCategories()
static opennlp.maxent.GISModel	train(DocumentCategorizerEventStream eventStream) Trains a new model for the DocumentCategorizerME.

More Resources

- Apache Tika – Java library for extracting structured text from most formats
  - http://lucene.apache.org/tika/
- Apache Lucene – Java library for indexing and searching text
  - http://lucene.apache.org/java/docs/