# Java - Chinese Ontology Library API

## Prerequisite

- JavaSE-1.7
- UTF-8 File Encoding

## EHowNet

### Load EHowNet Library and EHowNet Ontology

- Add `ontologyAcquisition.jar` to classpath

- Get an instance of the Ontology file `ehownet_ontology.txt`

```
EHowNetTree tree = EHowNetTree.getInstance("./docs/ehownet_ontology.txt");
```

### Search

- For example, we search for「開心」

```
List<EHowNetNode> results = tree.searchWord("開心");
EHowNetNode node = results.get(0);
```

- If there's no result, an empty List will be returned

### Data within a Node

- `node.getNodeType()` : return `NodeType.WORD` or `NodeType.TAXONOMY`
  - Node with type `NodeType.WORD` has no Hyponym, since it is at the bottom of the Ontology
- For word node:
  - `node.getSid()` : return an integer denoting the id of the word, for example `61549`
  - `node.getNodeName()` : return a string denoting the name of the word, for example `開心`
  - `node.getPos()` : return a string denoting the part-of-speech tag of the word, for example `Nv4,VH21`
  - `node.getEhownet()` : return a string denoting the ehownet's definition of the word, for example `{joyful|喜悅}`
- For taxonomy node:
  - `node.getNodeName()` : return a string denoting the name of the taxonomy, for example `物體`
  - `node.getEhownet()` : return a string denoting the ehownet's definition of the word, for example `object|物體`

### Hypernym

- `node.getHypernym()` : return an `EHowNetNode` instance, which is the parent of the node. If the node is at the top of the Ontology, the returned value will be `null`

### Hyponym

- `node.getHyponymList()` : return a `List<EHowNetNode>` instance, containing all the children of the node. If the node is at the bottom of the Ontology, an empty List will be returned

## CKIP Document Converter

### Convert a Text File into CKIP-Tagged Document

- Add `ontologyAcquisition.jar` and `jsoup-1.9.2.jar` to classpath

- Set the input/output files and convert

```
Converter.toCKIP("ckip_input.txt", "ckip_output.txt");
```

- We can also convert the documents online: http://sunlight.iis.sinica.edu.tw/uwextract/demo.htm

## Ontology Acquisition

### Load the Acquisition Tools

- Add `ontologyAcquisition.jar` and `jxl.jar` to classpath

- Initialize and start with root concept, CKIP-documents and EHowNet

```
OntologyAcquisition oa = new OntologyAcquisition("課綱", "./docs/ckip", "./docs/ehownet_ontology.txt");
oa.start();
```

### Search for a Specific Concept

- For example, we search for「會議」

```
OntologyNode node = oa.searchConcept("會議");
```

- If the concept does not exist, `null` will be returned

### Data within a Node

- `node.getConcept()` : return a string denoting the name of the concept, for example `會議` and `記錄`
- `node.getAttr()` : return a `List<String>` instance, containing all the related concept(but not Hypernym or Hyponym) of the node. If the node has no attributes, an empty List will be returned

### Hypernym

- `node.getHypernym()` : return an `OntologyNode` instance, which is the parent of the node. If the node is at the top of the Ontology, the returned value will be `null`

### Hyponym

- `node.getCategories()` : return a `List<OntologyNode>` instance, containing all the children of the node. If the node is at the bottom of the Ontology, an empty List will be returned

### Term/Document Frequency

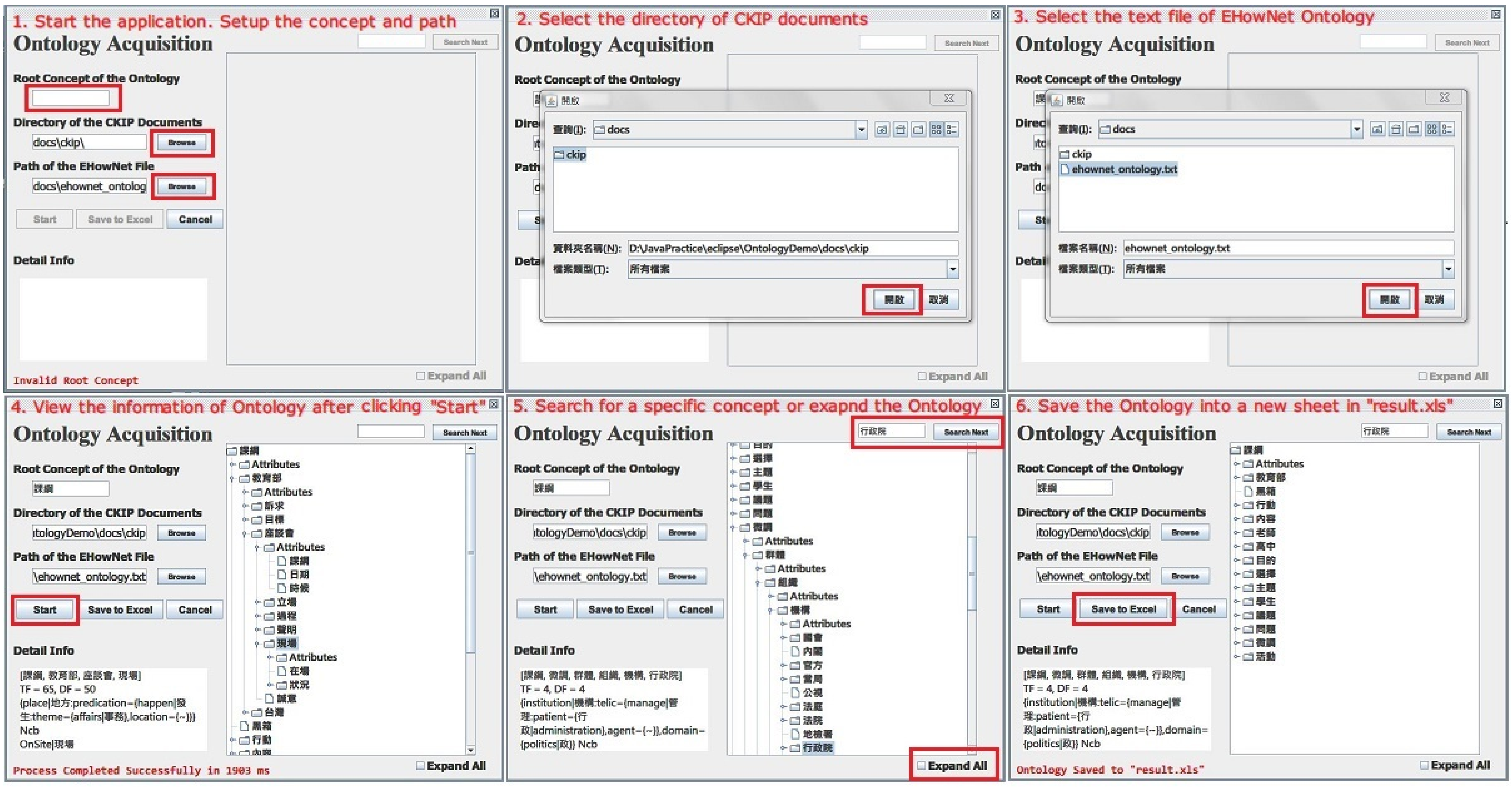
- `oa.getTermFreq("教育")` : return an integer, which is the term frequency of `教育`
- `oa.getDocFreq("教育")` : return an integer, which is the document frequency of `教育`

### Save the Ontology

- `oa.dump()` : save the Ontology into a new sheet in `result.xls`

### UI Version

- `new UIFrame()`



1. Start the application. Setup the concept and path

Ontology Acquisition

Root Concept of the Ontology

Directory of the CKIP Documents: docs\ckip\

Path of the EHowNet File: docs\ehownet_ontolog

Invalid Root Concept

2. Select the directory of CKIP documents

資料夾名稱(N): D:\JavaPractice\eclipse\OntologyDemo\docs\ckip

檔案類型(T): 所有檔案

3. Select the text file of EHowNet Ontology

檔案名稱(N): ehownet_ontology.txt

檔案類型(T): 所有檔案

4. View the information of Ontology after clicking "Start"

Root Concept of the Ontology: 課綱

Directory of the CKIP Documents: itologyDemo\docs\ckip

Path of the EHowNet File: \ehownet_ontology.txt

Detail Info

[課綱, 教育部, 座談會, 現場]
TF = 65, DF = 50
{place|地方:predication={happen|發生:theme={affairs|事務},location={~}}}
Ncb
OnSite|現場

Process Completed Successfully in 1903 ms

5. Search for a specific concept or exapnd the Ontology

Search: 行政院

Detail Info

[課綱, 微調, 群體, 組織, 機構, 行政院]
TF = 4, DF = 4
{institution|機構:telic={manage|管理:patient={行政|administration},agent={~}},domain={politics|政}} Ncb

6. Save the Ontology into a new sheet in "result.xls"

Detail Info

[課綱, 微調, 群體, 組織, 機構, 行政院]
TF = 4, DF = 4
{institution|機構:telic={manage|管理:patient={行政|administration},agent={~}},domain={politics|政}} Ncb

Ontology Saved to "result.xls"

## Ontology Doc2Vec

### Load the Tools and Build the Model

- Add `ontologyAcquisition.jar` to classpath

- Build the model with domain concept, CKIP-documents, EHowNet and dimension of the output vector

```
Doc2Vec d2v = new Doc2Vec("課綱", "./docs/ckip", "./docs/ehownet_ontology.txt", 5);
VectorModel model = d2v.build();
```

### Features and Valid Dimension

- `model.getFeatures()` : return a `List<String>` instance, denoting the features extraced by the model. An empty list will be returned if the process fails
- `model.getDimension()` : return an integer equal to the valid dimension

### Vectors

- `model.getDocVectors()` : return a `Map< String, List<Double> >` instance containing all the document vectors. `key` is the absolute path of a document while `value` is the vector
- `model.getDocVector("docs/ckip/97815.txt")` : return a `List<Double>` instance denoting the vector of the document with path `docs/ckip/97815.txt` . Both path and absolute path are acceptable for the parameter

## Compile and Run the Sample Project

- `OntologyDemo` is an Eclipse sample project of EHowNet, CKIP-Converter, Ontology Acquisition and Doc2Vec
- For Eclipse:
  - Open the project in workspace
  - `Properties-JavaBuildPath-Libraries` : add all the JAR files in `libs`
  - `Windows-Perferences-General-Workspace` : set the text file encoding to `UTF-8`
- For Shell:
  - `Makefile` is available
    - `OntologyDemo$ make` to compile, `OntologyDemo$ make run` to run
  - Commands to Compile and Run

```
OntologyDemo$ javac -d bin -sourcepath src -encoding utf8 -cp libs/jsoup-1.9.2.jar;libs/jxl.jar;libs/ontologyAcquisition.jar src/Main.java

OntologyDemo$ java -Dfile.encoding=UTF-8 -cp bin;libs/jsoup-1.9.2.jar;libs/jxl.jar;libs/ontologyAcquisition.jar Main
```

## Reference

- JExcel
- JSoup
- CKIP Service
- EHowNet