Document History		
EDITOR	DATE	COMMENT
Britt Fitch	2/22/12	created document.
Britt Fitch	8/21/12	Updated to reflect recent refactoring related to ease of implementation.

Core Functions

1. install
2. training a model (TRAIN)
3. scrubbing cases (TEST)
4. processing publications
5. properties

Install

To complete setup you must execute **install.sh**. This will do the following:

1. create a database (default: 'scrubber')
2. create tables
3. create a database user (default: 'scrubber')
4. populate lookup tables
5. download external dependencies (cTAKES, Weka)

If you choose to recreate the lookup tables yourself there is helper code supplied for this task.

There is a script provide for this lookup_umls table that contains the subset of the UMLS used in this application however, due to licensing restrictions it DOES NOT contain the UMLS CUID. The provided script is found in **sql/insert_lookup_umls.sql** and is executed as part of **install.sh**

1. To recreate the **Lookup_umls** table
    a. Install a local instance of the UMLS
    b. Execute **sql/create_umls_tables_from_local_install.sql**

There is a script provided to re-process publications in the event you desire to recreate the **lookup_term_frequency** table manually. To do so, execute **processPublications.sh**. This assumes that you have provided the set of open access publications that you wish to process.
The pre-processed term frequencies that are used in the Scrubber by default were calculated from a randomly selected 10,000 publication subset of the open access publications and is provided in **sql/insert_lookup_term_frequency.sql** (By default this is executed as part of **install.sh**).

1. To recreate the **lookup_term_frequency** table
    a. Acquire the set of publications you wish to process
    b. Execute **processPublications.sh**

Training a model (TRAIN)

1. Supply cases in either Spin XML format or plain text in the appropriate directory as defined in the properties section.

2. By default a trained model has been supplied for you and is located in **/data/models/train.arff**
3. To generate a new train model, execute **train.sh.** This will overwrite the supplied model based on your provided training set.

Scrubbing cases (TEST)

1. Supply cases in either Spin XML format or plain text in the appropriate directory as defined in the properties section.
2. By default a test model has been supplied for you and is located in **/data/models/test.arff**
3. To generate a new test model, execute **test.sh**. This will overwrite the supplied model and evaluate the new test model against the corresponding train model (located in **/data/models/train.arff**).
4. Scrubbed output is stored in **/data/scrubbed/test/**

Processing publications

1. Download open access publication set from NLM
   a. http://www.ncbi.nlm.nih.gov/pmc/tools/ftp/
2. By default a random selection of 10,000 publications have already been processed and provided for you through **install.sh**
3. To reprocess publication, download the desired data set and execute **processPublications.sh**

Properties

There are many configurable properties defined in **scrubber.properties**.
Additional details are supplied in ScrubberProperties.java.

`MYSQL_ADMIN_USER`	Mysql super user username
`MYSQL_ADMIN_PWD`	Mysql super user password
`DB_DRIVER`	Database driver, default is mysql
`DB_NAME`	Normal db username, default is 'scrubber'
`DB_USER`	Normal db username, default is 'scrubber'
`DB_PWD`	Normal db username, default is 'scrubber'
`DB_URI`	DB connection string, default assumes localhost
`DIR_INPUT_HUMAN_ANNOTATIONS_TRAIN`	Directory containing human annotated files used for training
`DIR_INPUT_HUMAN_ANNOTATIONS_TEST`	Directory containing human annotated files used for testing
`HUMAN_ANNOTATIONS_IMPLEMENTATION`	Class used for processing human annotated files. (alternative classes are available in the same package.)
`DIR_INPUT_PUBS_XML`	Directory containing open access publications in XML format. Only necessary if re-processing

	publications
DIR_INPUT_PUBS_TXT	Directory containing open access publications in TXT format. Produced by processing XML publications.
DIR_INPUT_PUBS_PROCESSED	Directory containing TXT format open access publications with inline citations removed. Produced by processing TXT publications.
DIR_INPUT_TRAIN	Input directory for TRAIN cases
DIR_INPUT_TEST	Input directory for TEST cases
DIR_OUTPUT_TEST	Output directory for scrubbed TEST cases
DIR_MODELS	Directory containing the TRAIN/TEST models
FILE_MODEL_TRAIN	TRAIN model filename
FILE_MODEL_TEST	TEST model filename
UIMA_READER_IMPL_TRAIN	Implementation of UIMA file reader used for TRAIN (default is XML, alternative options available in the same package.)
UIMA_READER_IMPL_TEST	Implementation of UIMA file reader used for TEST (default is XML, alternative options available in the same package.)
UIMA_READER_IMPL_PUBS	Implementation of UIMA file reader used for PUBS (default is TXT, alternative options available in the same package.)
CLASSIFICATION_COST_MATRIX	Cost matrix used as input to the J48 classification algorithm
LOCALHOST_NUM_THREADS	Number of threads to execute simultaneously, default=2.