Machine Learning for Adsorption Isotherm Data Extraction

Scraper

* Explanation is one Jupyter Notebook named “article\_scraper.ipynb” for what each step is doing
* Script that actually runs is on “articles.py” and is run through Bash loop called “loop.sh”
* To change which DOIs are scraped change the text file on “loop.sh”
* The code to format the JSON files is on “JSON Files” Jupyter Notebook
  + Specific to each publisher
  + Can only be run once for each article or it will mess up the text
* These JSON files were used as text in the machine learning part of this project
* Text files with lists of DOIs: “

Classification using Machine Learning

* The Scikit Learn website gives examples for every algorithm and how to work with text data on classification problems
* “Text Classification.ipynb” (3 group classification code)
* “isotherm\_classification.ipynb” (2 group classification code)
* Labeling Documents: “DOI list for text classification.xlsx” (3 group w/ DOIs and labels), “datalabels.csv” (3 group w/ just labels), “datalabels.xlsx” (3 group w/ just labels)
* The basic way that the algorithms work is:
  + Split the text into training and testing
  + Fit and transform training data into numbers (where vocabulary is built)
  + Just transform testing data (any unknown words are not counted because computer does not know what they mean)
  + Fit the training data to each algorithm
  + Have the algorithm predict on the testing data
  + See how well the algorithm predicted on the testing data
* Multinomial Naïve Bayes: <http://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html>
* Logistic Regression: <http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html>
* Support Vector Classifier: <http://scikit-learn.org/stable/modules/svm.html>
* Decision Tree with AdaBoost: <http://scikit-learn.org/stable/modules/tree.html> (decision tree), <http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostClassifier.html> (AdaBoost), <http://scikit-learn.org/stable/auto_examples/ensemble/plot_adaboost_regression.html> (example)
* The best way I’ve found to implement the algorithms is by creating a pipeline
  + This is the link to the page on the Scikit Learn website with instructions on how to do that <http://scikit-learn.org/stable/modules/generated/sklearn.pipeline.Pipeline.html>
  + I would only create this for the Support Vector Classifier and Decision Tree with AdaBoost because they had the highest accuracies

Locating Isotherms

* Code in file names “locating\_isotherms.ipynb”
* I have gotten to the point of having a block of relevant sentences from one article
* This needs to be scales and the block of sentences should be parsed (there are some attempts at that at the bottom of the notebook)
* I mainly used Chemdataextractor and Natural Language Toolkit along with the API in the Adsorption Database at NIST
* Chemdataextractor: <http://chemdataextractor.org/docs/intro>
* Natural Language Toolkit: <https://www.nltk.org/>