Machine Learning for Patent Classification

David Black & Peter Ciccolo

Problem Description & Background

Our project is an application of machine learning technology to text classification on United States patents to automatically differentiate between patents relating to the biotech industry and those unrelated. Additionally, we attempted to further divide biotech patents into various groups, using k-means and Chi-Squared to automatically identify potentially interesting subcategories in the patents. Our end goal, which has not yet been fully achieved, is a system that automatically obtains recently filed patents from the USPTO, classifies them, and enters them into a patent database.

This project was done in collaboration with Professor Woody Powell of the Sociology department in order to further his research. Professor Powell has been analyzing trends in the modern biotechnology industry by looking at patenting behavior. However, up until now he has been gathering said patent data by hand, which has proven largely impractical. Having an automated program extract biotechnology patents will further his research by leaps and bounds, and allow him to build a map of the of the entire country's biotech industry in order to analyze subtle trends in innovation.

For our positive training examples, we had access to a collection of approximately 20,000 patents that were collected via a project undertaken by David Black two years ago. For our negative training examples, we had access to an effectively arbitrarily large number (about 880,000) of unclassified patents. Both sets had some degree of noise due to the collection method, which we determined to be less than 5% on the positive training examples, and less than 25% on the unclassified patents.

For each patent, we had access to the following data: patent number, title, abstract, dates of application and acceptance, inventors and inventors' towns of residence, assignees and their locations, and various patent office-assigned classifications.

Problem Methodology & Justification

Of the data we had available on each patent, we chose to use only the title, abstract, and assignee names for training. This is mainly because the other fields were too inconsistently available for us to have confidence in their usefulness. In addition, based on how the positive training examples were collected, several of the fields would be artificially biased. For example, a disproportionate number of the positive examples were collected from the Boston area.

In order to facilitate training on the patents, some preprocessing was required. This consisted of first removing the undesired fields, lowercasing all text data, removing punctuation, and substituting a number token for all numerical terms.

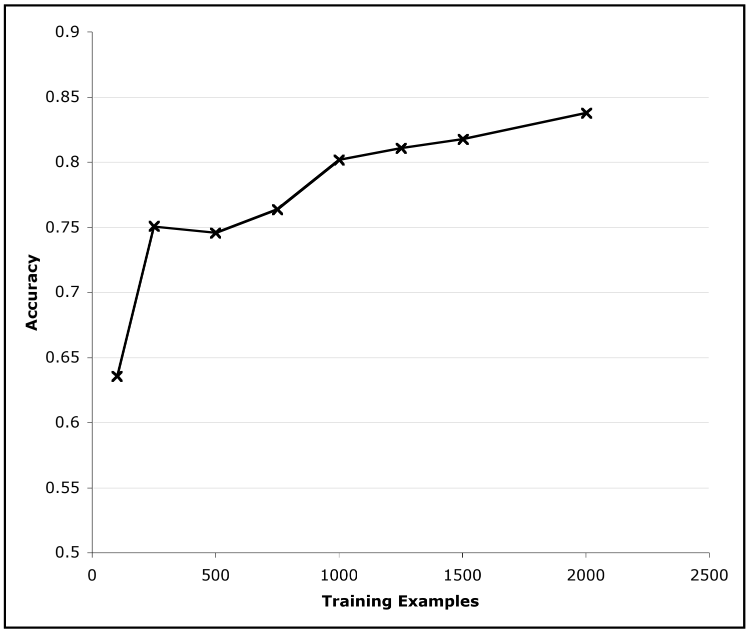
To create feature vectors from the data, additional processing was performed. First, the titles and abstracts were tokenized. Then the tokens were run through a Porter stemmer in order to collapse tokens into their semantic categories and reduce the feature count. The final token types used were assignees, title unigrams, title bigrams, abstract unigrams, and abstract bigrams. All tokens were tagged with their type. Finally, the vectors were length-normalized.

We then ran SMO to train an SVM. The SMO we implemented was based on Platt's 1998 paper with a few minor suggested improvements. The kernel used was Gaussian to account for the infinite dimensional input space. In addition, the SMO implementation used an error cache and dot-product cache in order to speed up execution. Since the SVM was linear, the final calculated product was a weight vector w and threshold b which could be saved to a file, eliminating the need to store all training examples.

Once classification results from the SVM had been obtained, the next step was to run a k-means clustering algorithm on those results in order to automatically discover potentially interesting subclasses within the biotech patents. Once these clusters were obtained, we ran a Chi-Squared analysis on them in an attempt to determine their semantic content.

Results & Conclusions

Due to time constraints, we were unable to run the SVM on the entire set of 40,000 training examples. However, training on even an extremely small dataset produced competitive results and there was consistent improvement in the accuracy as training set size increased.



Classification Accuracy vs
Training Set Size
(Learning Rate)

In addition to the data points shown on the graph, the SMO algorithm was run with 10,000 training examples. This resulted in a model with 93.7% accuracy over the entire dataset. SVM training time was the major obstacle encountered in this project, and future implementations would be well-served by efforts to improve SMO efficiency.

K-means divided the data into 4 groups, and Chi-Squared was used to discover the maximally discriminative features of these groups. Unfortunately, Chi-Squared analysis revealed that the groups discovered had little to no semantic content. This may have been partially due to the inclusion of common words in the feature set, and stop-word filtering might provide significant improvements, both to the value of k-means and the speed of SVM training.

In addition to the improvements described above, future plans for this project include training on the full dataset and setting up an automated harness to classify new patents drawn from the USPTO website as they are released.