NLP Final Project Proposal

Colin Pillsbury
Swarthmore College
Department of Computer Science
cpillsb1@swarthmore.edu

Tai Warner
Swarthmore College
Department of Computer Science
twarner2@swarthmore.edu

1 Project Description

For our final project we are interested in looking into a variety of approaches to the hyperpartisan news detection problem. One question that we are curious about is the possible benefits of performing dimensionality reduction and using more traditional supervised machine learning classifiers. Classifiers like Decision Trees and Logistic Regression might have trouble with input that has thousands of features, but could be useful if we are able to distill the features into a smaller subset. To perform dimensionality reduction we would look at the effectiveness of tools like PCA and LDA.

Another approach that we are considering is seeing if we could refine the labeling of the hyperpartisan articles. Because the training articles are labeled based on their news source, it's very possible that articles that come from ”hyperpartisan sources” are erroneously automatically labeled as being ”hyperpartisan” regardless of their content. To combat this issue, we are thinking of exploring using our classifier to reclassify articles, and then re-train on this refined training set.

Another area we were interested in was looking at combining classifiers, using Naive Bayes to predict based on the bag of words features and using another classifier to predict and some other set of features (e.g. number of ALLCAP words, whether or not the title contains a question mark). Using these two classifiers, we could take a weighted vote to predict whether or not a given articles is hyperpartisan.

2 Reading List

For learning about dimensionality reduction and feature selection in the context of NLP, we found a few articles that could be interesting. For example ”Text Categorization with Many Redundant Features: Using Aggressive Feature Selection to Make SVMs Competitive with C4.5” by Gabrilovich and Markovitch, and ”Dimensionality reduction for bag-of-words models: PCA vs LSA” by Ljungberg.

For relabeling data, we found some papers that tackle similar tasks, one of them being ”Identifying Mislabeled Training Data” by Brodley and Friedl.

Finally, for ensemble classifiers we did find a paper that refers to the use of ensembles in the context of sentiment analysis: ”Tweet Sentiment Analysis with Classifier Ensembles” by Hruschka, et al.

3 Data Set

For our tasks we would likely just use the provided training and validation data and use Spacy to preprocess the data.

4 Preliminary Results

The results on the following page are from early experiments using two classifiers: one is a dummy classifier that predicts the most frequent class and the other is a Multinomial Naive Bayes classifier. The classifiers were evaluated both using cross-validation and using the held-aside validation data.

5 Planned Methodologies

Because we haven't yet narrowed down to a single area that we want to explore, we don't have a concrete methodology planned yet. If we take the dimensionality reduction path, we could compare the performance of different dimensionality reduction techniques and also evaluate several types of classifiers (Decision Trees, Random Forests, SVMs, Boosting algorithms, etc.) For each of the models we could some hyperparameter tuning and then compare the results by using cross validation.

Model (evaluation	Accuracy	Precision	Recall	F1 Score
Most Frequent Class (cross validation)	0.500	0.000	0.000	0.000
Most Frequent Class (held-aside data)	0.500	0.000	0.000	0.000
Multinomial Naive Bayes (cross validation)	0.800	0.778	0.840	0.808
Multinomial Naive Bayes (held-aside data)	0.610	0.572	0.871	0.811

For the relabeling data approach, we could iteratively train a model, relabel data, and retrain a model on the new data. Each round we would evaluate the performance of the model by using a held-aside data set. With this approach we would probably not add many features beyond the regular bag of words features that we used earlier.

For the ensemble classifier approach, we could test a wide variety of combinations of classifiers to see which ensembles perform best. With this approach we would probably create some new features to go along with the bag of words features.