Problem Statement

Can we create a classifier ensemble that effectively generalizes to other datasets? The outcome of conducting this research will determine if a multi-model system is more efficient at generalizing to an unseen dataset than a standard one-model system.

Main Approach

To accomplish this goal, we propose a three-model system consisting of three different classifiers ___ ___ ___, each trained on their own dataset Amazon, IMDB, and Yelp accordingly. Each database will provide 1,000 randomly selected sentiments - 500 positive and 500 negative. From this, 80% of the data will be trained on their assigned classifier, and 20% will be held-out for the other two classifiers to train on. Essentially, each classifier will train 800 datapoints from their applicable dataset, and 400 on datapoints from the other two datasets. Once these classifiers are trained, we will gather the accuracies achieved during training. These accuracy scores will be used to weight the final prediction of the three combined models during testing, which is done on the Twitter dataset. The weights for each of the three classifiers during testing is calculated by their accuracy at train time softmaxed with the accuracies of the other two classifiers. During testing, each classifier will output a probability estimate for each of the two classes - positive or negative. For each classifier, we then multiply each class prediction by the classifier weight, sum the negative predictions, sum the positive predictions, and then make a final prediction based on the greater of these two sums.

Setup

- Data: The data will come from four different data sources: Amazon product reviews, IMDB movie reviews, Yelp restaurant reviews, and tweets from Twitter. We train our three different classifiers on the review datasets, and test on the tweet dataset.
- Task: The task is to create a multi-model (ensemble) classifier that generalizes to new types of data better than a baseline model.
- Metric: Accuracy, Precision, Recall, and F1 score will be used to evaluate our ensemble model as well as the baseline model.
- Baseline: The baseline for our ensemble classifier will be a ___ classifier trained randomly on only one of the three datasets, and tested on the Twitter dataset.

Work Plan (Schedule subject to change)

- March 21st - Parse and preprocess the data from the three datasets.
    - All three of us will devise a standard preprocessing technique
    - Once the standard is established, Bin will preprocess Amazon product reviews, Zane will preprocess IMDB movie reviews, Dillon will preprocess Yelp restaurant reviews, and one of us will preprocess the tweets from Twitter.
- March 28th - Finish designing the three classifiers plus the baseline classifier
    - Bin will design the ___ classifier, Zane will design the ___ classifier, and Dillon will design the ___ classifier.
    - All members will help create the baseline ___ classifier.
- April 4th - Classifiers are then trained on their datasets

    - Bin will train the ___ classifier on the Amazon dataset, Zane will train the ___ on the IMDB dataset, and Dillon will train the ___ on the Yelp dataset.
    - From each model, we pickle the model itself and the accuracy score of training.
- April 8th - Results from training are gathered and testing is prepared and conducted
    - All three will meet to test the multi-model ensemble classifier as well as the baseline model.
- April 15th - Evaluate the multi-model classifier against the baseline
    - Write-up results

Related Work

A comparative study of ensemble learning methods for classification in bioinformatics
https://ieeexplore.ieee.org/abstract/document/7943141
In this study, the researchers use bagging, boosting, and stacking to create a model ensemble for classification. During evaluation, accuracy is calculated based on the number of correct percentage of classifiers and root mean squared error. Like our proposal, this study utilizes multiple datasets for training/testing, however, our proposal differs such that we interleave the data from different datasets and train each model on a different dataset. Then, during evaluation, we weigh each model's output during testing based on the softmax of their accuracy achieved during training. We believe that our methodology will lead to a multi-model system that can more easily generalize to a new, unseen datasource. What's interesting is that they note that one model performs better than the other two models, however, they do not account for this in their final prediction during testing as we do. They concluded that their multi-model system worked much more efficiently than their baseline one-model system.

Tweet sentiment analysis with classifier ensembles
https://doi.org/10.1016/j.dss.2014.07.003
This study utilizes both a bag-of-words and feature hashing representation to create a model ensemble for classifying sentiment analysis using tweets from Twitter. The research links to another research study that lists three reasons for using an ensemble based system: statistical (different classifiers combined for a final output), computational (relieves the issue of a global optima that may be an issue for a single model), and representational (certain tasks may be difficult for only a single model to perform). This study, too, combines the final output using a summation of the polarity of the class and class probabilities to make a final prediction. Our approach differs slightly, since we're training and testing our models using different corpora at training time and combining the results in a novel way as well. The study proved conclusive that a multi-model system performed better than a standard single model system, however, we believe since we're testing our model on a different, unseen datasource, it will perform better than their ensemble model.

Ensemble of feature sets and classification algorithms for sentiment classification
http://nlpr-web.ia.ac.cn/cip/ZongPublications/2011/2011.02%20Information%20Science,%20XIA%20Rui.pdf

This study explores using an ensemble of Part of Speech (PoS) tagging and Word Relation (WR) based feature sets being trained on a Naive Bayes classifier, a maximum entropy classifier, and support vector machines (SVMs). These feature sets and classifiers are then combined using fixed combination, weighted combination, and meta-classifier combination. The paper then compares the performance of various combinations of these elements on five sentiment analysis data sets. Our goal is slightly different, as we aim to evaluate the flexibility of a certain model when applied to different genres of data, but the study is similar to ours in that it initially combines various models to achieve superior performance when compared to one individual model. This study also concluded, like the one above, that ensemble models generally perform than their single model counterparts, but it did not cover what we want to explore, which is how ensemble models trained on certain data generalize to other genres of data.