Classifying User Behaviors Across Domains

SUNet IDs: stevenfu, normanyu, agarg5
Names: Xiaofei Fu, Norman Yu, Abhishek Garg

By turning in this assignment, I agree by the Stanford honor code and declare that all of this is my own work.

1 Introduction

User profiling is a challenging and important task to online service provides. It can help generate advertisements that are targeted to users' specific interests. All service provides are already doing some type of user classification but their ability to do so is limited by the data they are from their own site and they will not use publicly available data from other channels. We are using data from many channels to improve user classification.

The question that we are trying to answer in this project is: given a review from Amazon, eBay or Twitter for a particular product, can we predict the Amazon hierarchical location of the product? As a more concrete example, if we are looking at a review for a music product on Amazon, can we train a classifier to predict whether it will be classified as classical, rock or R&B on Amazon?

The way we will see how well we did is test the accuracy of our predictions within a dataset and between datasets.

Predicting the exact location in the hierarchy is extremely challenging, so instead our goal will be to predict the depth-1 level categorization (e.g. classical, rock, R&B) in a particular product tree (e.g. music) All of our product reviews from all sources will be in from the same tree. With a model trained with labeled Amazon reviews, we then test it's effectiveness on review texts from Twitter and eBay, and try to improve the cross domain prediction results through various means.

2 Dataset

Our dataset was review texts with category labels that come from 3 channels: Amazon, Twitter and eBay. Thus our features will be extracted from the reviews.

Our targets will be the category ID given to the review. The data from Amazon comes with the category ID that is used by Amazon, and the reviews from eBay and Twitter were hand labeled with Amazon category IDs for the training and testing purpose.

As mentioned in the introduction, we decided to "rollup" product categories to their depth-level 1 categorizations. In our example, we keep 25 depth level 1 categories for music products, and rollup all category IDs to be the top of the tree it falls into.

We applied the same "rollup" strategy on Twitter and eBay data to find the top layer category this product (hence the review text) belongs to.

In the end of raw data processing, we parsed all of our raw data and generated our train/test input table file with only two columns per row, first column is the full review text, and second row is the top layer product category ID that associated to it. for example:

file	Review text	category ID
amazon_music.dat	I have been listening to this all day and never seem to tire of it. I saw the movie at an early screening a week ago and this is a great companion to the film experience. My hubby is waiting until he sees the film so he won't be spoiled on the exact songs included. As a child of the 70's I really love these songs...not in an ironic sort of way. In fact, I'm in the midst of a vinyl mid-life crisis so I have already ordered that version as well. I loved the movie and I love this soundtrack!	42
twitter_music.dat	Brett Rossi: Greatest Hits of the Baroque: http://t.co/HpGkodF2IH	85
ebay_music.dat	I bought my Calvin Klein Euphoria perfume because I had sampled it in a magazine and loved it. Euphoria makes me feel just like it's name. Apparently it makes other people feel that way too. I was at a gas station when the man on the other side of the pump came over to tell me he loved my perfume! I will certainly continue to buy Euphoria. Sybil	289122

After brief analysis of the processed data, we found that Amazon review text tended to be very long, and included a handful of features that we could choose from. So we decided to first focused on training the model primarily on Amazon data to test the model on other sources. Also we noticed that most of the "review" text from Twitter is merely the product name and a link the product page on Amazon. This would have a huge impact on our later strategy over Twitter data.

3 Features

We put a lot of effort into feature engineering for our project.

Our group had no prior experience with NLP, we we stuck to bag of words features. We were able to use term frequency - inverse document frequency (TF-IDF) to enhance the accuracy of our classifier. TF-IDF is a statistic on a word in a document that attempts to reflect how important a word is given a corpus of documents. It considers the *term frequency* which is basically the raw frequency of a term in a document as well as the *inverse document frequency*, which measures how rare the word is in the document.

We also applied basic NLP cleaning techniques to enhance the feature set. We found that not stemming words could potentially lead to overfitting as the algorithm would find the small differences between the stemming groups and fit on that noise.

The biggest thorn in our project was cross domain classification for Twitter tweets. A typical tweet was extremely terse, was often cutoff mid-sentence and usually allocated most of its characters to a link to a full product review. We applied several methods to try to specifically expand the Twitter data.

First, given our corpus of Twitter reviews, we knew that tweets with similar hashtags were usually referring to similar things. Thus, we attempted to expand our tweets by including words from other tweets with the same hash tag as features. We made sure to distinguish words that were in the original tweet from ones that we had aggregated from other tweets. We also tested over a small set of hyper-parameters for how many expanded words to include as well as a uniform weight to associate with each of the expanded words. As we expected, the accuracy did not increase monotonically with the number of words to include from expanded tweets nor did the accuracy increase monotonically with the uniform weight associated with the expanded words.

Next, we also attempted to use data from external sources to expand our Twitter tweets. The external data source that we used was Wordnet, which is a lexical database of the English language that has grouped words into synonym sets. Our approach was to take each word in the tweet and append all of the synonyms of the word to the tweet. The rationale for why this would improve our Twitter training accuracy is that we would 'hit' on more words that were found in the Amazon or eBay reviews.

4 Models

We used several different classifiers for our project. As a baseline we used multi-class logistic regression and varied the hyper parameters over the L1 and L2 penalties.

Our next approach was using support vector classification. We attempted to use several different kernels (linear, polynomial, radial basis function and sigmoid). Surprisingly we found that the linear basis function provided the best results.

Lastly, we also explored ensemble methods such as Random Forest. We found that the hyper parameter search for this algorithm was very hard to train. Very often the model would be severely overfit.

5 Results

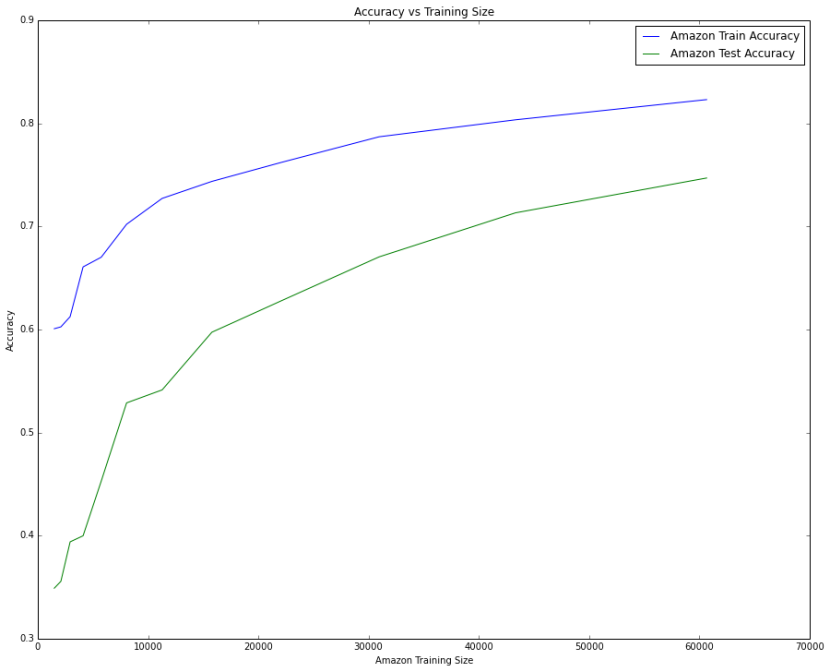
To analyze our results, we attempted to understand the marginal impacts of varying learning algorithm and feature engineering independently. First for learning algorithms, we found that varying learning algorithms can give a wide range of results for this problem. This is related to the sparsity of our data - for our Amazon dataset we had about 60k reviews and about 35k tokens, however for particular Amazon, there may only be on the order of 100 tokens that have non-zero value. Using linear SVM yielded significantly better results than any other learning algorithm we tried.

Train \ Test	Amazon	Ebay	Twitter
Amazon	80.9%	30.1%	25.0%
Ebay	20.6%	55.8%	22.1%
Twitter	14.1%	14.9%	63.7%

We were able to achieve Twitter specific increases in accuracy by using hashtag expansion. Using hashtag expansion increased the accuracy from 25% to about 31%, which is 20% improvement of the original result. However, we found that this technique is very data set specific. For example, this technique worked well for the music category on Amazon, but it did not work well for other categories, such as electronics or video games.

6 Discussion

First we'd like to understand whether or not there was enough training data for our model to do well.



The first figure shows the training and testing accuracy for Amazon for a classifier trained on Amazon. It in unclear how much having more data would have helped as the accuracy continued to improve as we used more data to train. Similar for the twitter accuracy, having more data may also have helped.

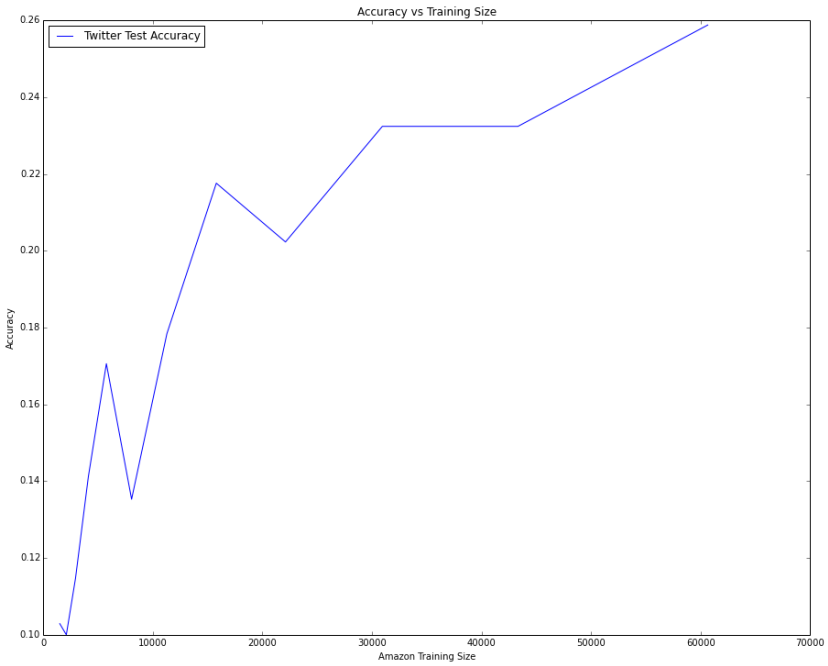
Throughout the course of the experimentation, we had trouble understanding why the Twitter training accuracy was so low. To help us better understand what sort of errors our classifier was making. The following plots show the confusion matrices testing Amazon for a classifier trained on Amazon as well and testing Amazon for a classifier trained on Twitter.

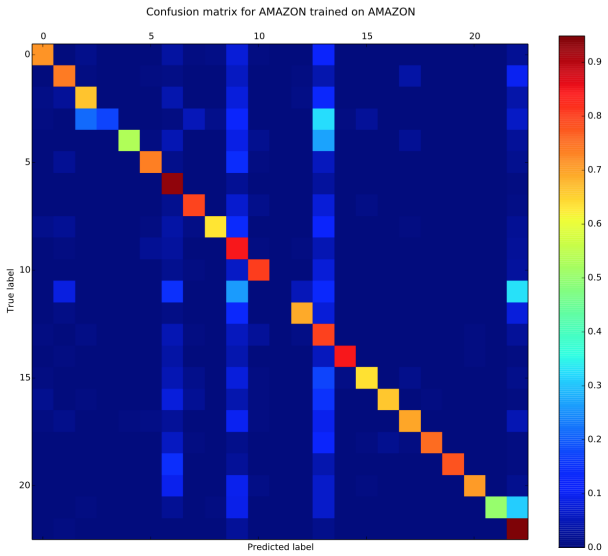
We can see that the classifier was able to predict Amazon using Amazon data well. On the other hand, the classifier was not able to predict Amazon using Twitter data well.

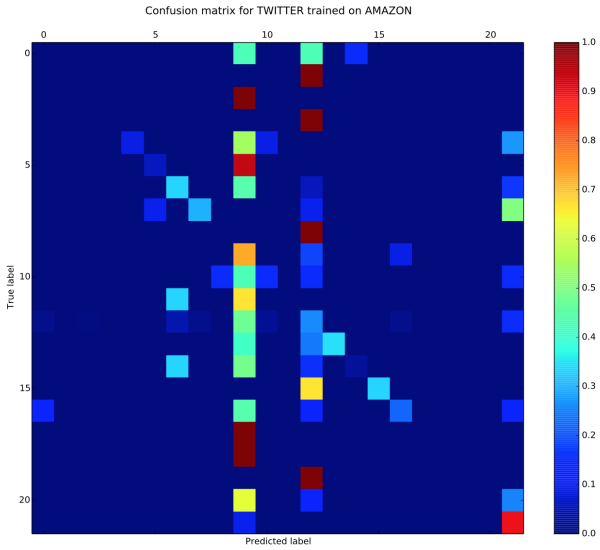
As we know for the testing results. The Amazon classifier was able to do a great job and there is a strong classification rate along the diagonal. However, there is a much different story for our results when we test on Twitter. In the Twitter trained on Amazon classifier, what we see is that the model is tends to predict two of the dominant labels. Those labels are the categories that are the most frequent in the Amazon data set. What this is telling us is that the model does not pick up on many strong features in the Twitter dataset the classifier relies mostly on its prior knowledge of the distribution on labels to do its classification.

7 Conclusion

The task of classifying reviews across domains is very challenging. We saw a significant drop off in performance in accuracy when we looked within domains and between domains. This reveals that the way people write reviews on different platforms is very different. However, we still believe that there is value to be gained for companies to incorporate information from other sources when attempting to learn more about their users.







8 Future

Our project focused on different ways we could improve classification by expanding text. We used external datasources (like WordNet) as well as internal data sources (using the tweets with the same hashtags) to try to increase the accuracy of our classifier and were able to achieve positive incremental gains. However, there appear to be limitations to this approach and other methods beyond basic ML NLP should be applied. An interesting next step would be to use latent Dirichlet allocation to develop a topic model for reviews and tweets and use the distribution over the topics as features to try to classify a review. Another approach would to explore more with feature engineering when using tweet expansion from other sources.