Transfer Learning on Sentiment Analysis

Nasrin Baratalipour

1 Introduction

Sentiment Analysis is defined as the automatic detection and classification of opinions expressed in the text written in natural language (Wiegand et al., 2010). In sentiment classification the polarity of a given text at the document or sentence level will be classified as positive, negative, or neutral.

With the widely-varying domains, researchers and engineers who build sentiment classification systems need to collect and acurate data for each new domain they encounter. However, annotating new domain is a costly task since the researchers should hire some annotators and spend money and time to annotate new domain document, while they have such big annotated data in other domains. In classification task sometimes we want to classify the data on one domain of interest, but we only have sufficient training data in another domain of interest, where the latter data may be follow a different data distribution. In such cases, ***knowledge transfer*** can improve the quality of the classification model for the small labeled data. This project focuses on applying the ***transfer learning*** on ***sentiment polarity classification***. Transfer learning can save a significant amount of labeling effort by adapting a classification model that is trained on some products to help learn classification models for some other products.

In this work we explicitly address the domain transfer problem for sentiment polarity classification by TrAdaBoost for transferring knowledge from one domain to another by boosting a basic learner. The basic idea is to select the most useful diff-domain instances as additional training data for predicting the labels of same-domain techniques. The theoretical analysis shows that TrAdaBoost first obeys the same-domain training data, and then chooses the most helpful diff-domain training instances as additional training data. Moreover, in our experiments, TrAdaBoost also demonstrates better transfer ability than traditional learning techniques. For this purpose we focus on online reviews for two different types of products, book and electronics.

In the next section we briefly review the role of machine learning in sentiment classification. Section (3) explains transfer learning and describes TrAdaBoost. Sections (4),(5) describe datasets, baseline and the experiments method. Section (6) explains the implementations of this project in detail. Finally we explain completely how to run and redo the experiments in section (7).

2 Machine Learning in Sentiment Analysis

Sentiment polarity classification is a classification task of labeling an opinionated text with two opposing classes (Positive vs. Negative) and sometimes neutral. In general, sentiment classification approaches are categorized into two groups:

1. Rule-based approach: in this approach the rules are created manually. These methods have the advantage that they do not require training data; thus it can be applied to any data set where training data are not available.

2. Machine Learning (ML) methods: In this approach, several features are defined for representing the text span. These features are supplied to a classifier and let the classifier to decide about the polarity of each text.

Our focus in this project is on Sentiment Classification based on Machine Learning approach, that we explain its related issues in follow.

2.1 How to present the text for ML approach

In the semantic analysis task, typically the bag-of-words model is used to represent a text. The most common bag-of-words model is binary unigram based representation of texts, which models document by presence or absent of words. These models can be expanded to n-grams of texts (e.g. bigrams) in order to model syntactic structure.

2.2 How to learn a sentiment classifier model

A classification model is needed to make the decisions about the classifier output based available data. There are different approaches in machine learning provide us a model.

- Supervised Approach
  The rise of the widespread availability to researchers of organized collections of opinionated documents was a major contributor to a large shift in direction toward supervised learning approaches. (Pang et al., 2002) compared multiple supervised machine learning algorithms (Naive Bayes, maximum entropy classifiers, support vector ma- chines) for the task of sentiment classification of movie reviews.

  In all supervised approaches, reasonably high accuracy can be obtained subject only to the requirement that test data be similar to training data. To move a supervised sentiment classifier to another domain would require collecting annotated data in the new domain and retraining the classifier. This dependency on annotated training data is one major shortcoming of all supervised methods.

- Unsupervised Approach
  Unsupervised approaches to sentiment classification can solve the problem of domain dependency and reduce the need for annotated training data.

An unsupervised system iteratively extracts positive and negative sentiment items, which can be used to classify documents. However, most academic papers suggest that unsupervised methods will not perform as well as supervised methods.

By considering the pros and cons of both supervised and unsupervised learning, we are interested to find a way to enable us to use the existing annotated corpora without having the problem of domain dependency.

2.3 Domain Consideration

The accuracy of sentiment classification can be influenced by the domain of the items to which it is applied. One reason is that the same phrase can indicate different sentiment in different domains: for example where "go read the book" most likely indicates positive sentiment for book reviews, but negative sentiment for movie reviews; or consider that "unpredictable" is a positive description for a movie plot but a negative description for a car's steering abilities. Difference in vocabularies across different domains also adds to the difficulty when applying classifiers trained on labeled data in one domain to test data in another.

Several studies explore different approaches to customizing a sentiment classification system to a new target domain in the absence of large amounts of labeled data. (Yang et al., 2006) took the following simple approach to domain transfer: they find features that are good subjectivity indicators in both of two different domains (in their case, movie reviews versus product reviews), and consider these features to be good domain-independent features.

(Blitzer et al., 2007) explicitly address the domain transfer problem for sentiment polarity classification by extending the structural correspondence learning algorithm (SCL), achieving an average of 46% improvement over a supervised baseline for sentiment polarity classification of 5 different types of product reviews mined from Amazon.com.

If we find a solution for the problem that how to accurately classify the new test data by making the maximum use of the old data, we can apply it on sentiment domain adaptation. In the next section, we are talking more about transfer learning and one of the well-defined method in this approach.

3 Transfer Learning

A fundamental assumption in classification learning is that the data distributions of training and test sets should be identical. When the assumption does not hold, traditional classification methods might perform worse. However, in practice, this assumption may not always hold. For example, in Web mining, the Web data used in training a Web-page classification model can be easily out-dated when applied to the Web some- time later, because the topics on the web change frequently. Often, new data are expensive to label and thus their quantities are limited due to cost issues. How to accurately classify the new test

data by making the maximum use of the old data becomes a critical problem. Although the training data are more or less out-dated, there are certain parts of the data that can still be reused. That is, knowledge learned from this part of the data can still be of use in training a classifier for the new data.

In (Dai et al., 2007) to enable transfer learning, they use part of the labeled training data that have the same distribution as the test data to play a role in building the classification model. They call these training data same-distribution training data. The quantity of these same-distribution training data is often inadequate to train a good classifier for the test data. The training data, whose distribution may differ from the test data, perhaps because they are out-dated, are called diff-distribution training data. These data are assumed to be abundant, but the classifiers learned from these data cannot classify the test data well due to different data distributions.

3.1 TrAdaBoost

TrAdaBoost is Transfer AdaBoost learning framework, which extends AdaBoost for transfer learning (Dai et al., 2007). AdaBoost is a learning framework which aims to boost the accuracy of a weak learner by carefully adjusting the weights of training instances and learn a classifier accordingly. However, AdaBoost is similar to most traditional machine learning methods by assuming the distributions of the training and test data to be identical. In the proposed extension to AdaBoost(TrAdaBoost), AdaBoost is still applied to same-distribution training data to build the base of the model. But, for diff-distribution training instances, when they are wrongly predicted due to distribution changes by the learned model, these instances could be those that are the most dissimilar to the same-distribution instances. Thus, in our ex- tension, we add a mechanism to decrease the weights of these instances in order to weaken their impacts.

As noted, the main problem in sentiment classification is the lack of new domain for learning a good model and do sentiment classification in the new domain based on the learned model. So, in this project we investigate the effect of TrAdaBoost as a method in transfer learning model in sentiment classification. In the next section, we explain the dataset we used to evaluate the presented approach and the baseline and gold approach to compare the output results. In the next section of the review, we call domain adaptation instead of using transfer learning in sentiment classification. In addition, we use the term of "target-domain" for the dataset in short supply and which is in the same-distribution with test(target) data. we also use the term of source-domain for the dataset which are available in the large number but they are in the diff-distribution with test(target) data.

4 Experiment

4.1 Dataset

In this project we used Multi-Domain Sentiment dataset (Blitzer et al., 2007) for the sentiment domain adaptation. Multi-Domain Sentiment dataset contains product reviews taken from Amazon.com from several product domains. In this dataset, some domains (e.g. *books*) have hundreds of thousands of reviews while others (e.g. *musical instruments*) have only a few hundred. In this corpus, each review consists of a rating (0-5 stars), reviews with rating > 3 were labeled positive, those with rating < 3 were labeled negative, and the rest discarded because their polarity was ambiguous. After this process, 1000 positive and 1000 negative examples remained for each domain (Blitzer et al., 2007).

From this dataset, we chose the DVD section of the dataset as the target-domain and the Book section and the Elec. section as the source-domain. To simulate the lack of enough instances in the target-domain, we re-sampled the dataset so that it fits to the transfer learning scenario. In the re-sampling, we was careful to keep data balanced in two positive and negative categories. At the end we have created five dataset incorporated in training the baseline, two different experiments, training the gold system, and testing all the models. The description of the each dataset used for each phase is summarized in the Table 1.

	DVD Reviews (Target-Domain)	Book Reviews (Source-Domain)	Elec. Reviews (Source-Domain)
Total	2000	2000	2000
Baseline	100	-	-
Exp. 1.	100	-	2000
Exp. 2.	100	2000	-
Gold	1700	-	-
Test	300	-	-

Table 1: dataset specifications

4.2 Baseline and Gold System

The baseline is a classifier trained on the small portion of target-domain data without adaptation. we put this experiment in my work, as an estimation for the model created by the small labeled data. So, with this experiment we can understand how a model can be strong when it is learned just on the 100 labeled data in the DVD domain.

However, the gold standard is a classifier trained on the all reviews in the target-domain. we do this experiment to know what is the maximum accuracy that we can get from the learning algorithm when there is enough data to train a model.

4.3 Domain-Adaptation

Since the number of labeled data is rare in some domains and it is expensive to create the data in the domain which is needed to use in sentiment classification, we proposed to apply TrAdaBoost approach(Dai et al., 2007), the method in transfer learning, to re-use the data in other domain(source-domain) in order to classify the data in the rare domain(target-domain). For this purpose, we did two separated experiments in order to simulate the effect of the data in different source-domains for making the classification model for a target-domain data. In the first experiment, we took Electronic reviews as the source-domain and in the second experiment we took Book reviews for the source-domain. For both experiment DVD reviews are used as the target-domain.

More specifically as it is shown in the Table 1 in the first experiment, we used 100 reviews in the target-domain(DVD) and 2000 reviews in the source-domain(Elec.) for building the classification model based on TrAdaBoost approach. In the second experiment, we used 100 reviews in the target-domain(DVD) and 2000 reviews in the source-domain(Book) for building the classification model based on TrAdaBoost approach.

For testing all the experiments, including experiment1, experiment2, baseline and gold standard, we used 300 reviews randomly selected from DVD domain.

4.4 Evaluation

As we mentioned in the previous section, in the experiments, we use Support Vector Machines as the basic learners in both AdaBoost and TrAdaBoost approaches, with the same setting.

We use accuracy as a measure for evaluating TrAdaBoost approach on different domains and comparing it with the traditional AdaBoost approach.

$$accuracy = \frac{\#truepositives + \#truenegatives}{\#truepositives + falsepositives + falsenegatives + truenegatives}$$

As you can see in accuracy formula both categories, positive and negative, have the same values. The main reason for choosing this measure is that in our classification task negative and positive category worth same and also the used data set is balanced.

5 Results[1]

As it is shown in the Table 1, we used 100 reviews from DVD domain for training the baseline model, while in the gold standard 2000 reviews used for learning the model. We used *AdaBoost* classifier and *SVM* classifier for both baseline

[1] we got some new results during the week after the presentation, the results reported here are the new results.

and gold standard approaches. However it is found in my experiments that AdaBoost can hardly improve the generalization error of SVM on all the data sets, so we just presented the results obtained by AdaBoost.

For evaluating TrAdaBoost we have defined the experiment1 and experiment2. In the experiment1, we compare the result of applying AdaBoost versus TrAdaBoost approach on the DVD reviews as the target-domain data and electronic reviews as the source-domain data. The process in the experiment2 is the same as experiment1, with this difference that the the source-domain data in this experiment is changed to book.

Figure 1 shows the results of experiment1 and its corresponding baseline and gold standard. For the baseline, we trained an AdaBoost model on the 100 provided training data and got the accuracy of 64.66 when we tested the model on the test data with 300 reviews. For the gold, we trained an AdaBoost model on the 1700 labeled DVD data and got the accuracy of 79%, which is the maximum accuracy that we can get if there is enough annotated data. we also mixed up the target-domain(100 DVD reviews) data and source-domain(2000 Elec. reviews) data and run AdaBoost and TrAdaBoost on this new train data. For AdaBoost we observed that the accuracy improved to 69.33, there are some features(words) in the Elec. reviews, which help the model learned for classifying the data in DVD domain. In the second run for TrAdaBoost the results improved more and the accuracy increased to 74%. So, The accuracy given by AdaBoost on the same train data in this experiment, are strictly lower than those given by TrAdaBoost approach. So it means that TrAdaBoost by smart re-weighting schema improves the model learned for classifying the rare labeled data.

However, as several researchers have noted, transfer learning does not always improve the results. As it is shown in the Figure 2 in the experiment2, when we changed the source-domain to the book reviews, the accuracy in the TrAdaBoost drops one percent compared to traditional AdaBoost. While the accuracy increased by 11 percent in the AdaBoost compared to the baseline, which means that the number of common general words between book domain and DVD domain are significant. So, we can get the maximum improvement in the traditional AdaBoost without any smart re-weighting got from TrAdaBoost.

6 Conclusion

In this project we applied transfer learning in sentiment classification task. We used TrAdaBoost as an approach for making a learning model in the domains with the small number of labeled data, which is no enough for making a strong learning model. So, TrAdaBoost by introducing a new boosting approach for weighting the train data instances improves the model learned for classifying small labeled data. We made some different experiments and figured out this approach improves the model for the domains which are related together.

As a feature work of this study, we can expand source-domains. In this work, we only considered two domains for source-domain. However, we can

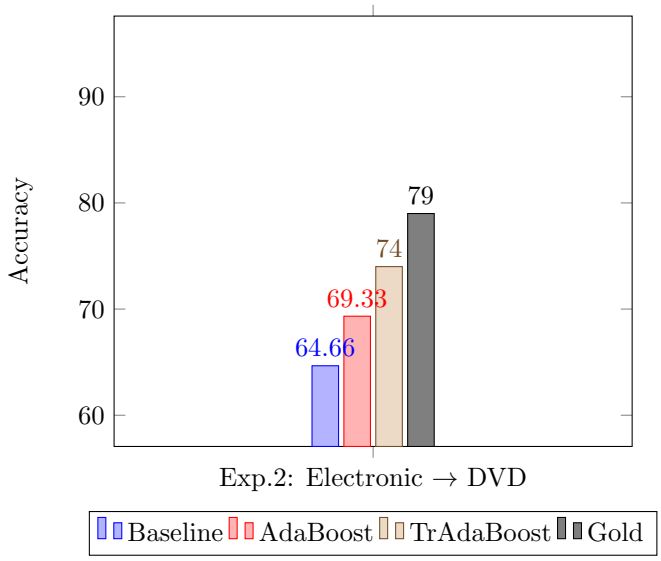

Figure 1: Comparing the different accuracies obtained on Elec. reviews as the source-domain and DVD reviews as the target-domain data

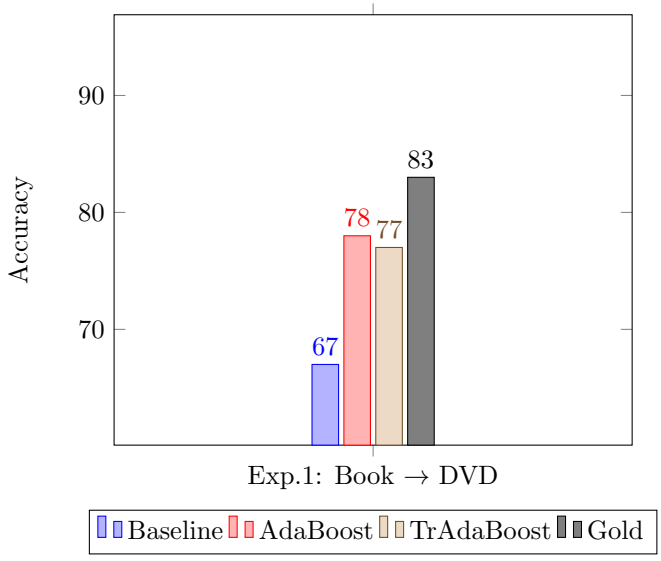

Figure 2: Comparing the different accuracies obtained on Book reviews as the source-domain and DVD reviews as the target-domain data

expand those domain to all the available domains in the dataset. Then, the similarity and differences between effective source-domains (source-domains that

significantly increase the performance of the system after using trAdaBosst) and non-effective source domain can be investigate.

There another factor in the transfer learning that we did not consider in this study, which is the size of target-domain dataset. The smaller the size of target-domain, the more effective the transfer learning is.

7 Implementation details

7.1 Preprocessing

We used bag-of-words model for representing each review in a binary feature set. So, the features of our model are the words represented in the document. For this purpose, we consider each word as an individual feature and put the word present in the sentence as the value of the feature. If the word occurs in the sentence, the feature representing this word get the value of 1, otherwise it get 0.

However, with this approach the number of features becomes very large and we cannot learn a classification model based on this huge amount of features without very large train set. Hence, we made a heuristic to remove the features which their representative words have seen less than 10 times in the whole the corpus. The main reason behind of this heuristic is that if a word happens less than 10 times in the corpus this amount is not enough to build the model based on. Also, if we want to keep this huge amount of data in the features set, we need to have a bigger dataset to train a model based on these feature correctly. With this simple heuristic the number of features in each domain reduced from around 15000 features to 5000 features.

7.2 TrAdaBoost

For TrAdaBoost we used the code presented here:
`http://www.cse.ust.hk/TL/index.html`
On this webpage you can find the variety of sources created for transfer learning task. For this project we used the first one based on the paper we was interested to work on: *boosting for transfer learning* (Dai et al., 2007).
Since we was interested to use TrAdaBoost approach for sentiment classification task, it was needed to change the input dataset of the program. However, the code was written and worked only on the special dataset which was used in the paper. So, we read the code carefully and did make some changes in the code and make it possible to work with Amazon reviews (Blitzer et al., 2007). You can find the code in `C_TraDaBoost` folder in the root directory.
Finally, we made two bash files to make running the program and redo my experiments easy.

8 How to Run

The structure of zip file after extracting is shown in Figure 3. In the Data folder, there are two folders namely 'exp1' and 'exp2' contain the experiment 1 and experiment 2 dataset respectively. In each of these folders, we have two folders that contain the raw dataset (e.g. 'dvd' and 'electronics' for experiment 1), the 'trAdaBoost' folder contains files used in learning trAdaBoost and finally the 'weka' folder contains the weka convention files. The trAdaBoost and weka convention files can be generated from the raw text with the java codes in the 'Data converter' folder. We named the files so that they can be easily distinguish from others with respect to the table 1. For example the dataset used in the training the Gold system is 'dvd_1700.arff' and the one used in learning the baseline system is 'dvd_100.arff' (c.f to the next sections for more details on how to run baselines and gold system).

In the next section, we explain how to use WEKA in order to get the results of baseline and gold standard. Then, in the last section, we will explain the way that the TrAdaBoost can be run.

8.1 How to run baseline

Please follow the following steps to learn AdaBoost model on the target-domain datasets and and test the learned model:

1. First open WEKA and then go to Applications → Explorer. Open the file `data/exp1/weka/dvd_100.arff`
2. Go to: Classify → Choose → Meta → SMO
3. Now select Supplied test set, and upload the test `data/exp1/weka/dvd_300.arff`
4. Click on start
5. You are done.

8.2 How to run the gold system and the AdaBoost system

As mentioned, the only difference of baseline and gold is in the size of training data. Therefore, the way that you can get the gold results is exactly the same as baseline, just in the first step, you need to upload this file: `data/exp1/weka/dvd_1700.arff`.

Similarly to learn and test the AdaBoost system, you only need to alternate the learning file in step 1 and choose `data/exp1/weka/electronics2k_dvd100.arff` or `data/exp1/weka/book_2000-dvd100.arff` for experiment 1 and experiment 2 respectively.

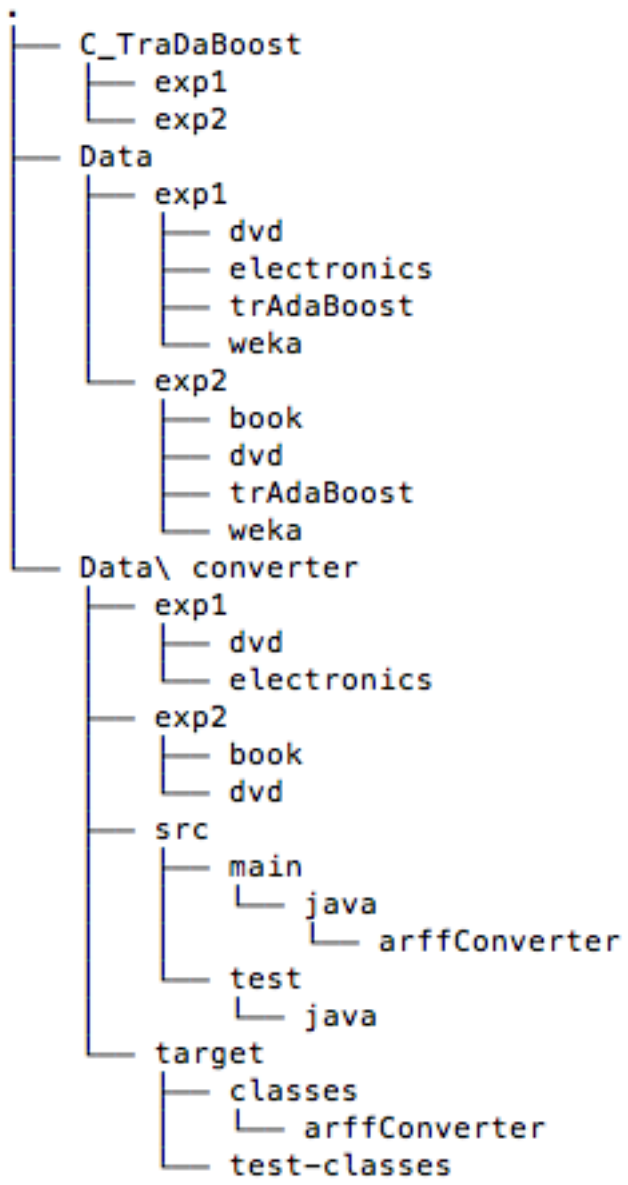

Figure 3: The structure of folders after extracting the zip file.

8.3 How to run TrAdaBoost

As mentioned before, for TrAdaBoost we used the C code presented in (Dai et al., 2007) and we made some minor changes in the code for making it possible to work with the dataset that we proposed to work on for this project (Blitzer et al., 2007). More specifically, the codes do not allow to use a fix test dataset and always randomly select the test dataset from source-domain dataset. We changed the C++ codes so that always used the defined dataset for all target-domain, source-domain and test dataset.

To make running the TrAdaboost more easier, we copied the data from 'Data' folder to the 'C_TradaBoost' folder, namely 'exp1' and 'exp2'. To work with the code, you just need to run the bash files separated for two different experiments in a windows platform.

- exp1.bat
  This runnable file allows you to run TrAdaBoost on experiment 1 dataset. The result of the TrAdaBoost run will be store in the `C_TradaBoost/exp1/results-exp1.txt`.

- exp2.bat
  This runnable file allows you to run TrAdaBoost on experiment 2 dataset. The result of the TrAdaBoost run will be store in the `C_TradaBoost/exp2/results-exp2.txt`.

References

John Blitzer, Mark Dredze, Fernando Pereira, et al. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *ACL*, volume 7, page 440–447, 2007.

Wenyuan Dai, Qiang Yang, Gui-Rong Xue, and Yong Yu. Boosting for transfer learning. In *Proceedings of the 24th international conference on Machine learning*, page 193–200. ACM, 2007.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, page 79–86. Association for Computational Linguistics, 2002.

Michael Wiegand, Alexandra Balahur, Benjamin Roth, Dietrich Klakow, and Andrés Montoyo. A survey on the role of negation in sentiment analysis. In *Proceedings of the workshop on negation and speculation in natural language processing*, pages 60–68. Association for Computational Linguistics, 2010.

Hui Yang, Jamie Callan, and Luo Si. Knowledge Transfer and Opinion Detection in the TREC 2006 Blog Track. In *TREC*, 2006.