# Domain Adaptation in Sentiment Analysis of Twitter

**Viswa Mani Kiran Peddinti and Prakriti Chintalapoodi**

University of Southern California, Los Angeles, CA, USA. 90007

peddintiviswamani@gmail.com, c.prakriti@gmail.com

## Abstract

This paper focuses on performing Sentiment Analysis of Twitter by adapting data from other domains, commonly referred to as Domain Adaptation. While we show that Domain Adaptation is useful in predicting sentiments, we propose different techniques to select an out-of-domain data source that would aid in Sentiment Analysis. Additionally, we suggest two iterative algorithms based on Expectation-Maximization (EM) and Rocchio SVM that filter noisy data during adaptation and train only on valid data. Finally, we explore a couple of metrics, Mutual Information and Cosine distance to measure similarity between different domains of data. We use Twitter and Blippr as data sources and perform binary sentiment (positive and negative sentiments) classification.

## Introduction

Twitter is a micro-blogging service that has been become popular due to its data volume and availability in mobile devices. The real-time nature and avalanche effect caused due to re-tweeting makes it an informative source for potentially changing trends (of products, events, reviews etc.). This lucrative feature attracts researchers and developers to build different business intelligence applications like classifying debates [13], e-governance [Clarie-2006;Namhee-2006], earthquake detection [20]. Thus, this "publish-and-subscribe" social network [13] provides directed links among users and hosts emotion rich information across wide set of users making it a potential source for sentiment analysis applications.

Sentiment Analysis has been a topic of interest since a long time; however, it recently became increasingly popular for usage with micro-texts like twitter; partly due to large user data available with them and partly due the challenges it poses in analyzing them. The fundamental challenge comes from brevity of the text [1] which leads to non-standard text artifacts and de-contextualization [21], making it harder for sentiment prediction. This inherent problem is not limited to machines but also for human labeling, making sentiment a subjective measure. Apart from these, diversity among users and their styles [19] makes the data noisy. Averaging such noise using statistical systems requires large human-labeled training data, which is often very costly. This motivates use of additional, cheap and readily available labeled data from other sources, suggesting the use of Domain Adaptation.

Domain Adaptation is a critical requirement in Sentiment Analysis of Twitter, not only because it addresses labeled data problems, but also because it brings new features from the new domain to the target domain, especially in applications where the features change with time. "Movies" is one such domain where important features (words related to movies/movie names) change with time and Domain Adaptation improves the models trained from stale data by importing all the new features from the new domain. Thus an old model built using human labeled stale in-domain data can be made ready to classify current data by using Domain Adaptation techniques to import data from readily available labeled data from other domains.

Despite its importance, Domain Adaptation is a less explored technique for Sentiment Analysis of Twitter mostly due to its limitations. In accord to "Domain Transfer Problem" [18], features that highly correlate to sentiment labels in Twitter are often not found in other domains and vice-versa due to the unique expressive style of tweets called "web-grammar". However, clever techniques can be used to filter the features that have high correlation with Twitter labels and use them to enhance current models. In this paper we propose such algorithms and techniques that make Domain Adaptation possible for Sentiment Analysis of movie Tweets by adapting from IMDB and Blippr data. The choice of the source domain for adaptation plays an important role in enhancing the model and hence must be chosen wisely. For this, we propose metrics that not only indicate similarity of domains with in-domain (Twitter) but also guide in selection of parameters.

IMDB[1] is a popular Internet Database containing complete information about movies, their rankings and reviews for the movies provided both by professionals and users. Blippr[2] is a social networking site that enables users to share their views/reviews in the form of "blips" for different categories including movies. It differs from IMDB by limiting reviews to 160 characters.

[1] http://imdb.com

[2] http://blippr.com

The unique contributions of this paper are:

- Introduction of a new feature reduction technique that selects features having high correlation with target labels.
- Study the effect of Domain Adaptation during training on Twitter tweets.
- Data selection/filtering algorithms:
  (i) EM with a control parameter, SimFact
  (ii) Rocchio SVM.
- Two Metrics, Mutual Information and Cosine distance that indicate similarity between domains and their implications in selection of data points and parameters.

## Data and Pre-Processing

The movie domain is chosen for the experiments due to the rich emotional content in them. Twitter tweets form the in-domain data while IMDB and Blippr form the out-of-domain data for our experiments. Due to lack of a standard data set for Twitter, the tweets are collected manually using the Twitter API. Tweets are human labeled as positive, negative and neutral. We filter neutral tweets during pre-processing and perform Sentiment Analysis on tweets conveying only positive and negative sentiments. The pre-processing step in this case is a human filter as both the test and train data are human labeled. The train and test data sets comprise of 1735 and 192 tweets respectively from each category.

Blippr data is obtained similarly using its API where blips with score greater than zero are labeled positive while those with score lower than zero are labeled negative. 2618 blips from each category form the train data set. IMDB movie reviews were obtained from [3]. However, only 2618 reviews were randomly chosen to make train set size same as that of Blippr.

We sanitize emoticons, usernames, hash-tags, urls and numbers as in [2] from the train, test and out-domain data. Since word n-grams are used as features, the dimensionality quickly increases with training data and choice of 'n'. Hence feature reduction is an important step to limit it to practical limits while not compromising on accuracy. One popular technique is 'thresholding' where the histogram's tail i.e. features with very less frequency or counts are removed. We introduce a new feature reduction technique called Relative Information Index (RII), which when combined with thresholding forms a good feature reduction technique. Formally, RII for an n-class problem can be defined as,

$$RII = \frac{\sum_i^n \sum_j^n \left| c_i - c_j \right|}{2 * \sum_i^n c_i}$$

where $c_i$ is the frequency count of the $i^{th}$ class.

Inspiration for RII comes from Pointwise Mutual Information, but the difference being, RII is calculated on each feature instead of a pair of features. RII whose value ranges from 0 to 1 can be interpreted as the amount of information the feature adds to classification. An RII value close to 0 indicates that the feature has same counts for all classes and hence does not contribute significantly to the final decision. The effect of RII based on number of features reduced and corresponding accuracy can be seen in Table-1. For evaluation, we use F-score (average of class F-scores) as an accuracy measure.

| | Original | Thresholding | RII | both |
|---|---|---|---|---|
| F-score | 0.694 | 0.69 | 0.844 | 0.844 |
| # features | 55440 | 3085 | 51964 | 3085 |

Table 1: Feature Reduction results for 3-class problem using NB

## Methodology

### Adaptation

To observe the advantage of using Domain Adaptation, we form our baseline by training on strict in-domain (twitter) data and test on the same domain. We use Naïve Bayes (NB) and SVMs for classification, where trigram features with both Threshold (count 1) and RII (threshold=0.1) are used. Naïve Bayes experiments were performed using Weka while SVMLight was used for SVM. We additionally explore the use of normalization of feature vectors by using weak filters. The baseline results including effect of normalization can be seen in Table 2.

For Domain Adaptation, we add out-domain (IMDB and Blippr) data in different percentages of the total out-domain data available to the in-domain and record the improvements in Table-3.

### Data Selection

For filtering noise and data selection, we propose two algorithms: Expectation Maximization and Rocchio SVM. The motivation for these methods is that data points from out-domain data that are not supported by the model trained from in-domain can be considered as outliers or noisy data points.

For the first approach, we use a variant of Expectation Maximization (EM) algorithm called Feedback EM (FEM) to iteratively move from the initial in-domain model by slowly consuming valid data points from out-domain. In place of unlabeled data, we use out-domain data with their labels for FEM. We use the initial model to predict the labels for the out-domain data and the data points which are correctly labeled are considered as the ones the initial model can explain and hence close to initial data distribution. Next, we move the model to include these data points by taking partial counts, as probability is indicative of how much the model can explain the data point. The algorithm is iterated until likelihood associated with out-domain domain converges. To ensure that the model does not deviate much from the initial in-domain model, every iteration involves re-training on in-domain data. To prevent over-learning as the out-domain points selected are very similar to original model, the model is re-estimated on these points. Since we take partial counts from mislabeled data only, we call this "Hard-FEM". To encounter over-learning, we develop "Soft-FEM" where weighted partial counts are collected not only from the correctly labeled data points but also from misclassified data points. The weight factor is called "Similarity Factor" (SimFact), where correctly classified and misclassified data points' partial counts are weighted by SimFact and 1-SimFact respectively. It can be noted that for a SimFact of 1, Soft-FEM is equivalent to Hard-FEM. Again, it is important to note the significance of the SimFact, which takes a value from 0-1. It implies a prior on how similar the data distributions are. For distributions that are extremely similar, misclassified data points need to be given higher weightage to improve the model, hence SimFact must be close to 0. For distributions that are very different, misclassified data are noisy data and hence must be given low weightage; hence SimFact must be close to 1. We experiment with complete in-domain and out-domain data with different SimFact values and the results can be seen from Table -3.

For the second approach, we use a method inspired by Rocchio, an early text classification method [16]. We apply the Rocchio algorithm in order to detect samples in the source domain that are most similar to the target domain samples in two phases. Building a Rocchio classifier is achieved by constructing a prototype vector $C_j$ for each class j as follows:

$$C_j = \boldsymbol{\alpha} (M_j) - \boldsymbol{\beta} (M_k)$$

where $M_j$ = normalized mean vector of class j
$M_k$ = normalized mean vector of class k

In our problem, we construct prototype vectors for the source and target domains considering class j to comprise of Twitter (target) samples and class k to comprise of IMDB or Blipper (source) samples. α and β are parameters that adjust the relative impact of samples from classes j and k. [5] recommends α = 16 and β = 4. Cosine similarity is used to measure the similarity between every source domain sample and the prototype vectors. If this distance is greater than a threshold, this sample is included in the "reliable source domain samples" set R. All samples in R can be treated as samples that are not Twitter-like.

Now an SVM is trained treating target samples as positive class, samples in R as negative class and tested on samples in source domain not included in R. This classifier is iterated by adding the samples classified as negative to the set R. The samples that the final SVM classifies as positive are selected from the source domain to adapt with the target domain to perform better sentiment analysis.

The algorithm to perform sample selection is described below:

1. Assign the target domain samples to class j and source domain samples to class k.
2. Construct prototype vectors for classes j and k.
3. For each sample d in class k,
   If [sim(d, $M_j$) - sim(d, $M_k$)] > threshold,
   R = R U dAssign target domain samples as positive (set P) and samples in R as negative. Set Q = samples om source domain not in R.
4. Use P and R to train a SVM classifier SI, with i = 1 initially, i = i+1 with each iteration (line 5-7)
5. Classify Q using Si. Append samples in Q classified negative into R in next iteration.
6. Use final SVM classifier to classify source domain samples.

### Adaptability Metrics

An important challenge for Domain Adaptation is choosing the right domain and similarly choosing the SimFact for the FEM data selection process. For these, we propose two adaptability metrics that indicate how close the domains are. We compute the metric score on IMDB-Twitter pair and Blippr-Twitter pair where the features for each pair are derived from the combined training set.

For the first metric, we use Mutual Information (MI) to measure the additional information the out-domain brings to the in-domain distribution. A value close to zero indicates that the distributions are independent and hence very different. A positive MI indicates that distributions are dependent and hence similar. Table-5 shows MI values for the IMDB-Twitter and Blippr-Twitter pairs.

Another popular distance measure used was Cosine Distance (CD), which gives the length of the projection of a point with another.Cosine distance was measured between the mean of in- and out-domain data. Longer projection indicates, smaller angle thus higher similarity between domains. Table-6 shows Cosine Distance for the domain pairs.

## Results and Discussion

We start our discussion of results with the effectiveness of the RII feature reduction technique introduced in the earlier section. Unlike other experiments, we perform this experiment with a 3-class problem of positive, negative and neutral classes and use Naïve Bayes (NB) classifier with trigram features. Table-1 results indicate there is a 94% reduction in feature dimensions and F-score has improved by 0.154 (22.2%). This is a significant result as

RII has boosted the F-score but caused reduction in number of features, while thresholding did not improve F-score though it reduced the number of features significantly. Hence, using both the feature reduction techniques is found to be beneficial. We use the RII technique for rest of the paper for a binary class problem.

| | NB | NB(norm) | SVM |
|---|---|---|---|
| F-score | 0.646 | 0.83 | 0.773 |

Table 2: Base Line results for complete in-domain training

Table -2 results show the Baseline results using NB and SVM where normalizing the feature values help because of the inherent variability present in the length of the data/features of data from Twitter, IMDB and Blippr. Also as expected, SVM results in a better F-score than NB.

Domain Adaptation with varied percentages of the total out-domain data available was observed to determine the ideal ratio of in-domain to out-domain. Table-3 results indicate that IMDB performs better than Blippr while Table-4 indicates that Blippr performs better than IMDB. This variation in results comes from the fact that Blippr and IMDB have different Data lengths. Blips, like Tweets are limited in number of characters and hence have much lower feature counts compared to IMDB movie reviews. Hence normalizing the feature domains will yield results where Blippr performs better than IMDB due to its similarity with Twitter as users use same kind of "web grammar". Table-5 shows results for using SVM. SVM has showed clear boost in the F-score by using Domain Adaptation. From all three tables, one can also observe the variation of F-score across different % of out-domain data due to the amount of noise introduced by out-domain data. All three methods agree that 80% out-domain data yields better results. This percentage measures to out-domain to in-domain ratio of 1.21, which can be termed as approximate ideal ratio.

| percent | IMDB | Blippr |
|---|---|---|
| 10% | 0.635 | 0.64 |
| 20% | 0.641 | 0.648 |
| 30% | 0.645 | 0.659 |
| 40% | 0.654 | 0.665 |
| 50% | 0.665 | 0.662 |
| 60% | 0.662 | 0.648 |
| 70% | 0.673 | 0.662 |
| 80% | 0.678 | 0.662 |
| 90% | 0.669 | 0.658 |
| 100% | 0.666 | 0.652 |

Table 3: NB results for Domain Adaptation with different % of out-domain data

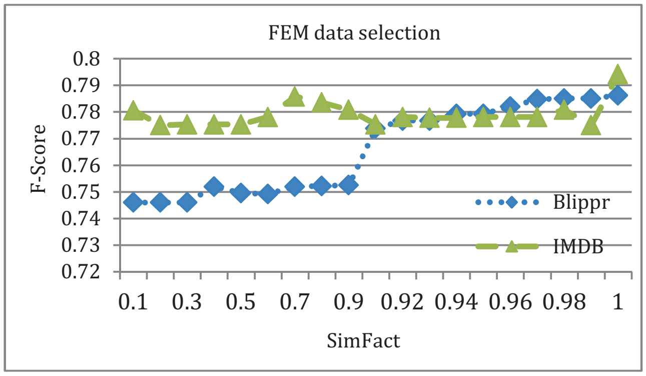
| percent | IMDB | Blippr |
|---|---|---|
| 10% | 0.803 | 0.811 |
| 20% | 0.791 | 0.811 |
| 30% | 0.781 | 0.806 |
| 40% | 0.774 | 0.805 |
| 50% | 0.789 | 0.808 |
| 60% | 0.778 | 0.814 |
| 70% | 0.771 | 0.811 |
| 80% | 0.771 | 0.812 |
| 90% | 0.77 | 0.823 |
| 100% | 0.772 | 0.823 |

Table 4: NB (norm) results for Domain Adaptation

| percent | IMDB | Blippr |
|---|---|---|
| 10% | 0.864 | 0.803 |
| 20% | 0.869 | 0.814 |
| 30% | 0.871 | 0.820 |
| 40% | 0.874 | 0.823 |
| 50% | 0.881 | 0.833 |
| 60% | 0.894 | 0.833 |
| 70% | 0.897 | 0.843 |
| 80% | 0.904 | 0.849 |
| 90% | N/A | N/A |
| 100% | N/A | N/A |

Table 5: SVM results for Domain Adaptation with different % of out-domain data

Graph-1 shows the F-score accuracy with using FEM algorithm for different SimFact values as a data selection algorithm. The ideal SimFact values for both IMDB and Blippr are close to 1 indicating a good enough measure for Domain Adaptation. However the F-score accuracy of IMDB is much higher than Blippr because of the large information provided by IMDB and lack of normalization of feature vectors.



Graph 1: EM accuracy for different SimFact values

| Threshold | % samples chosen | | F-Score | |
|---|---|---|---|---|
| | Blippr | IMDB | Blippr | IMDB |
| 0.05 | 39.09% | 50.21% | 67.65 | 62.23 |
| 0.005 | 42.33% | 53.06% | 68.71 | 63.11 |
| 0.0005 | 45.87% | 54.68% | 69.18 | 64.62 |

Table 6: Results for Rocchio SVM.

Table 6 shows the results of using Rocchio SVM as a data selection algorithm. The percentage of the data found meaningful by this method indicates that IMDB has more information than Blippr, which again comes from the fact that IMDB reviews are longer in length and hence contain more information. However, comparing Graph-1 and Table-6, EM gives better results compared to the Rocchio SVM.

The similarity metric scores between IMDB-Twitter and IMDB-Blippr scores can be seen in Table 7. Mutual Information (MI) scores are positive indicating that both distributions have high similarity with respect to Twitter and can be used for Domain Adaptation. Additionally, the MI and Cosine Distance scores are higher for Blippr indicating greater similarity between the Twitter and IMDB domains. This result is expected due to the similar blogging style of Blippr with Twitter.

| Domain | Mutual Information | Cosine Distance |
|---|---|---|
| Blippr | 4.4408 | 0.8672 |
| IMDB | 1.4834 | 0.7477 |

Table 7: Metric Similarity between IMDB and Blippr

## Conclusion and Future Work

In this paper, we have successfully demonstrated that domain adaptation is a useful technique to aid Sentiment Analysis of Twitter and gives an F-score as high as 0.9 using SVM. We have also introduced a new feature reduction technique that not only reduces the number of features drastically but also enhances F-score when used with thresholding. The proposed data selection algorithms do a fair job in selecting the ideal data points and filtering the noise. The FEM based selection algorithm performs better than the Rocchio and suggests that Blipper performs better than IMDB. The metrics introduced are representative of the similarity between distributions and suggest that the similarity between Blippr and Twitter is higher than between IMDB and Twitter.

Thus, we perform Domain Adaptation for Sentiment Analysis of Twitter using a relatively small number of labeled Twitter tweets while the remaining data is obtained from IMDB or preferably Blippr (or any other suitable data source). An abstract idea can be obtained by computing the adaptability metrics to determine the ideal ratio of combination and also the SimFact required for the FEM data selection process.

This paper is a preliminary work towards Domain Adaptation for Sentiment Analysis in Twitter and still has scope for exploration and improvement. Despite the fact that Data selection algorithms perform better than Baseline systems, they still lag behind the highest attained accuracy by SVM using the heuristic selection of percentage of out-domain data, hence requiring improvement. Although the metrics positively suggest that Bipper is similar than IMDB, they do not quantitatively indicate the ideal ratio of out-doman to in-domain data or the selection of the value of SimFact.

## References

[1] A. Bermingham, et. al. 2010. Classifying sentiment in micro blogs: Is brevity an advantage?, CIKM ‘10, October 26–29, 2010, Toronto, Ontario, Canada.

[2] A.Go, et.al. 2009. Twitter sentiment classification using distant supervision, CS224N Project Report,Stanford.

[3] B.Pang, et.al. 2002. Thumbs up? Sentiment Classification using Machine Learning Techniques, In proceedings of the Conference on Emperical Methods in Natural Language Processing (EMLP), pages 7986.

[4] B. Pang, L. Lee. 2004. A sentiment education: Sentiment analysis using subjectivity summarization based on minimum cuts. In Proceedings of ACL-04,42nd Meeting of the Association for Computational Linguistics, pages 271–278.

[5] B. Liu. 2010. Sentiment analysis: A multi-faceted problem, Invited contribution to IEEE Intelligent System.

[6] J. Blitzer, et.al. 2007. Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification, Proceedings of the Association for Computational Linguistics (ACL).

[7] C. Buckley, et.al. 1994. The effect of adding relevance information in a relevance feedback environment, SIGIR94.

[8] H. Daume, D. Marcu. 2006. Domain Adaptation for StatisticalClassifiers, Journal of Artificial Intelligence Research 26.

[9] G. Li, et.al. 2010. Micro-blogging Sentiment Detection by Collaborative Online Learning, ICDM 2010, Sydney, Australia, December 13-December 17.

[10] J. Blitzer, et.al. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification, in Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, Prague, Czech Republic, pp. 432–439

[11] L. Zhuang, et.al 2006. Movie Review Mining and Summarization, CIKM '06: Proceedings of the 15th ACM

international conference on Information and knowledge management New York,USA: ACM, p. 43--50.

[12] X. Li, B. Liu. 2003. Learning to Classify Texts Using Positiveand Unlabeled Data, IJCAI-03.

[13] N. A. Diakopoulos, et.al. 2010. Characterizing Debate Performance via Aggregated Twitter Sentiment, CHI 2010.

[14] O. Chapelle, et.al. 2008. Optimization Techniques for Semi-Supervised Support Vector Machines, Journal of Machine Learning Research 9, 203-233.

[15] R.Collobert, et.al. 2006. Large scale transductive SVMs, Journal of Machine Learning Research,7:1687–1712.

[16] J. Rocchio. 1971. Relevant feedback in information retrieval,in G. Salton (ed.).

[17] S. Tan, et.al. 2009. Adapting Naive Bayes to Domain Adaptation for Sentiment Analysis, ECIR 2009, LNCS 5478, pp. 337–349.

[18] S. Prakash, S. Dhupar. 2010. Mining Sentiments from Twitter Posts. CS 548 Final Research Paper, USC.

[19] T. Sakaki, et.al. 2010. Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors, WWW2010, Raleigh, North Carolina.

[20] T. Mullen, N. Collier. 2004. Sentiment analysis using support vector machines with diverse information sources, In Proceedings of EMNLP-2004, pages 412– 418, Barcelona, Spain.

[21] V. Sindhwani, et.al. 2005. Beyond the point cloud: From transductive to semisupervised learning, International Conference on Machine Learning.

[22] W. B. Claster, et.al. 2010. Unsupervised Artificial Neural Nets for Modeling Movie Sentiment, 2010 Second International Conference on Computational Intelligence, Communication Systems and Networks.

[23] W. B. Claster, et.al. 2010. Thailand-Tourism and Conflict: Modeling Sentiment from Twitter Tweets using Naive Bayes and Unsupervised Neural Nets, CIMSim2010: Computational Intelligence, Modeling and Simulation.