Domain Adaptation in Sentiment Analysis of Twitter

Viswa Mani Kiran Peddinti Prakriti Chintalapoodi

University of Southern California, CA, USA

Abstract

- Sentiment Analysis (SA) requires large human labeled data; which is costly to obtain.
- Domain Adaptation(DA) techniques help in performing SA with minimum human labeled data.
- Two techniques, Feedback EM and Rocchio SVM are proposed for data selection/filtering.
- Use of Mutual Information(MI) and Cosine Distance(CD) to measure similarity between In and Out-Domain distributions.

Motivation

- Brevity of text, text artifacts, de-contextualization, subjectivity and diversity cause noisy data and labels.
- High cost associated with human labeling (averaging labels over multiple labelers).
- Dynamic domain features - E.g. Movie names change with time.
- Domain Adaptation Problem: Low correlation between one domain features and other domain labels.
- Maintaining integrity/style of In-Domain data upon adaptation.

Data Collection

- Human labeled Twitter data (In-Domain) with 1735 (train) + 192 (test) was collected for both positive and negative categories. Neutral tweets were discarded.
- 2618 blips were collected from Blipper (Out-Domain) API for both categories.Blip score of above zero is considered as positive and below zero as negative sentiment.
- IMDB reviews were obtained from [1]. 2618 reviews were selected randomly for each positive and negative categories.

Pre-Processing

- N-gram features scale quickly with large data and with higher 'N'
- Standard feature reduction techniques like PCA are costly and impracticable for large data sets.
- Features that occur too-sparse or too-frequent in all classes don't contribute to decision process.
- Sparse features are removed by 'Thresholding' - Remove features with "count=1"
- Relative Information Index (RII) is developed inspired from MI. However unlike MI, RII acts on one feature at a time.

$$RII = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}|C_i - C_j|}{\sum_{k} C_k},$$

where C_i = feature count for i^{th} class

- Features with similar counts for all classes have low RII and hence don't contribute to decision.

Methods

Adaptation

- Weka was used to perform the Naïve Bayes classification and SVMLite was used for SVM classifier.
- Trigram (N=3) features with thresholding (threshold=1) and RII (threshold=0.1) steps were used for pre-processing.
- The ideal ratio of In-Domain and Out-Domain data was measured by varying % of total Out-Domain data points, while fixing the no. of In-Domain data points.

Data Selection

Feedback EM (FEM)

- An iterative selection/filtering of Out-Domain data, consuming data that supports the previous iteration model and diversifying the current model to include only similar data points.
- Training involves updating the feature counts of positive and negative classes.
- Each iteration involves re-training on In-Domain data (without filtering) to prevent large deviation from the original model. Also to prevent over learning, updates were performed for only misclassified data.
- Selection/Filtering was performed by classifying the data points with the current model.
- Convergence of Out-Domain data likelihood is used as the stopping criterion for iterations.
- Limiting selection to points that are correctly classified by current model is restrictive, prominently in cases where the In-Domain and Out-Domain data are known to be similar.
- Two variations of FEM; Hard FEM - No partial counts from mislabeled Out-Domain data. Soft FEM - partial counts (factored by SimFact) for mis-classified Out-Domain data points.
- Similarity Factor (SimFact) represents the similarity between In & Out-Domain data.SimFact=1 - In & Out-Domain are similar/same. Simfact=0 - They are very different (Hard FEM).

Rocchio SVM

- Rocchio algorithm [6] was used to detect suitable points from Out-Domain data in two phases.
- As a first step, a prototype vector is constructed for each class.

$$C_j = \alpha(M_j) - \beta(M_k)$$

where M_j = Normalized mean vector for class j; M_k = Normalized mean vector for class k

- Cosine similarity is measured between each data point and prototype vector. For data points having value higher than threshold form the samples 'not similar to In-Domain'.
- Next, SVM is trained with In & Out-Domain samples as "positive" and "Negative" classes respectively.
- The classifier is iterated classifying the left-over samples, until no more changes are made to these sets.

Adaptability Metrics

- MI and Cosine distance between In & Out-Domain data was measured and related with adaptability of the Out-Domain data.
- We show the higher the similarity metric higher is the adaptability.

Results

Feature Reduction

- Threshold removes the long tail, thus gives a high reduction (94.4%) in features with slight deterioration in F-Score (-0.5%)
- RII removes the insignificant features and has relatively less reduction (6.86%) however obtains large improvement in the F-score (21.6%)
- The joint usage of RII and thresholding brings the best of both with an overall 94% reduction in features and 21.6% improvement in F-Score

	Original	Thresh	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 and NB classifier*

	NB	NB (Norm)	SVM
F-Score	0.646	0.83	0.773

Table 2: *Baseline results for complete In-Domain training*

%	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 DA*

%	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%	0.711	0.823
100%	0.772	0.823

Table 4: *NB (Norm) results for DA*

Domain	MI	CD
Blippr	4.4408	0.8672
IMDB	1.4834	0.7477

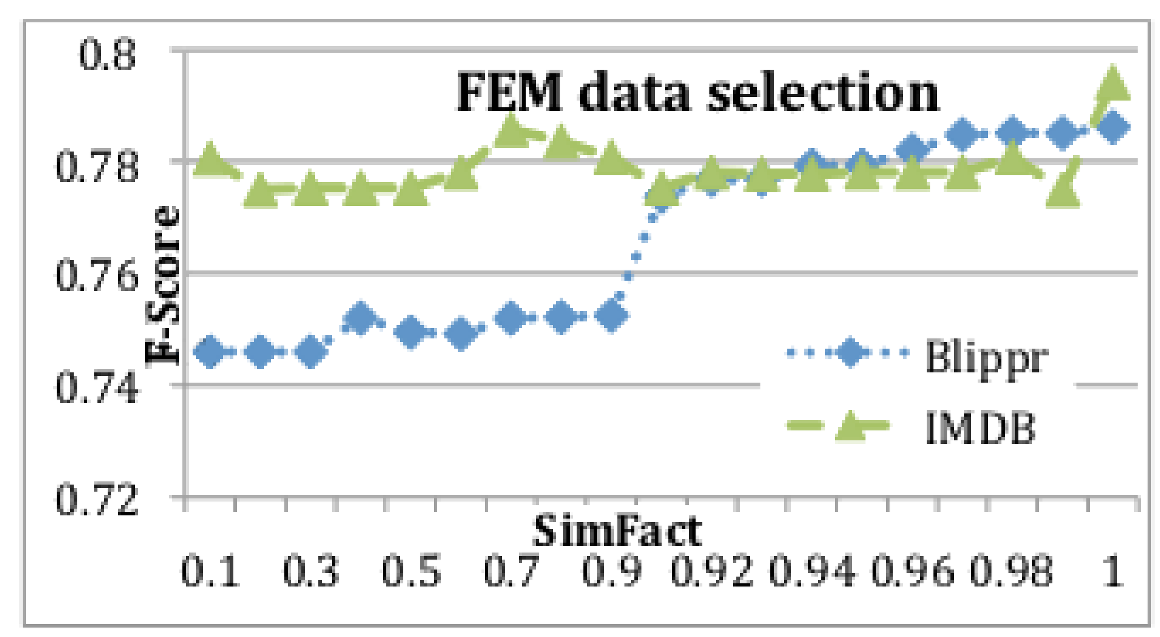
Table 5: *Metric Similarities between IMDB & Blippr*

Thresh	Samples chosen		FScore	
	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*

%	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

Table 7: *SVM results for DA*



Graph 1: EM accuracy for different SimFact

References

[1] B. Pang. et. al Thumbs up? Sentiment Classification using Machine Learning Techniques. *EMPL*, pages 7986, 2002.

[2] J. Blitzer , et. al Biographies, Bollywood, Boom boxes and Blenders: Domain Adaptation for Sentiment Classification *ACL*, 2007

[3] G. Li, et. al Micro-blogging Sentiment Detection by collaborative On-line Learning. *ICDM-10*, 2010.

[4] L. Zhuang, et. al Movie Review Mining and Summarization *CIKM'06*, 43-50, 2006.

[5] X. Li, B. Liu Learning to classify Texts Using Positive Unlabeled Data. *IJCAI-03*, 2003.

[6] J. Rocchio Relevant Feedback in information retrieval *G.Salton (ed.)*, 1971.

[7] S. Tan, et. al Adapting Naïve Bayes to Domain Adaptation for Sentiment Analysis. *EICR*, LNCS 5478: 337-349, 2009.

[8] T. Mullen, N.Collier Sentiment Analysis using support vector machines with diverse information sources. *EMNLP-04*, 412-418,2004.

Author contact: peddinti@usc.edu, prakritc@usc.edu