A Novel Scheme for Domain-transfer Problem in the context of Sentiment Analysis

Songbo Tan, Gaowei Wu, Huifeng Tang and Xueqi Cheng
Information Security Center, Institute of Computing Technology, Chinese Academy of Sciences, P.R. China
tansongbo@gmail.com, {tansongbo,tanghuifeng}@software.ict.ac.cn, {wgw,cxq}@ict.ac.cn

ABSTRACT

In this work, we attempt to tackle domain-transfer problem by combining old-domain labeled examples with new-domain unlabeled ones. The basic idea is to use old-domain-trained classifier to label some informative unlabeled examples in new domain, and retrain the base classifier over these selected examples. The experimental results demonstrate that proposed scheme can significantly boost the accuracy of the base sentiment classifier on new domain.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing; I.2.7 [**Artificial Intelligence**]: Natural language processing

General Terms

Algorithms; Performance; Experimentation

Keywords

Sentiment Classification; Opinion Mining; Information Retrieval

1. INTRODUCTION

With the rapid growth of semantic web page (such as product reviews) in Internet, the detection and analysis of opinions, feelings, or attitudes expressed in a text has attracted more and more attention in the community of information retrieval and natural language processing. A key problem in this area is sentiment classification, in which a document is labeled as a positive or negative evaluation of a target object (book, product, etc.). In most cases, the use of statistical or machine learning techniques has proven to be successful in this context, such as Naive Bayes (NB), Maximum Entropy Classification (ME), and Support Vector Machines (SVM)) [5].

Due to highly domain-specific nature, however, supervised sentiment classifier [1][2][4][5] typically requires a large amount of new labeled training data when moving from one domain to another (e.g. from "house reviews" to "computer reviews"). As a result, when transferred to a new domain without any labeled examples, a sentiment classifier often performs extremely bad. This is so-called domain-transfer problem [1][7][8].

CIKM'07, November 6--8, 2007, Lisboa, Portugal.

A simple solution to this problem is to manually label a large number of examples for each new domain. Unfortunately, however, this method is unfeasible in practice because the acquisition of these labeled data can be time-consuming and expensive. Consequently, it is an important and urgent job to investigate an ideal and practicable method for domain-transfer problem.

To the best of our knowledge, no previous work has been conducted on exactly this kind of problem, where there are a large amount of labeled data in old domain but scarcely any labeled data in new domain. The most related research is conducted by Aue and Gamon [1]. But their work still needs to label a small amount of training data for a new domain.

In this work, we attempt to tackle domain-transfer problem by combining old-domain labeled examples with new-domain unlabeled ones. The basic idea is to use old-domain-trained classifier ("old classifier" for brevity) to label top n most informative unlabeled examples in new domain and learn a new classifier based on these selected examples (n is a pre-defined number indicating how many examples in new domain shall be picked out as informative ones). Without loss of generality, we employ centroid classifier [3] as the base classifier.

2. BASE CLASSIFIER

In this work, the documents are represented using vector space model. In this model, each document d is considered to be a vector in the term-space. For term weight we employ TFIDF [6]. First we compute a centroid C_i using formula (1) for each class c_i:

$$C_i = \frac{1}{|c_i|}\sum_{d \in c_i} d \tag{1}$$

where $|z|$ indicates the cardinality of set z.

Then we count the similarity of one document d to each centroid by cosine measure,

$$Sim(d, C_i) = \frac{d \cdot C_i}{\|d\|_2 \|C_i\|_2} \tag{2}$$

where $\|z\|_2$ denotes the 2-norm of z, and "." denotes the dot-product of the two vectors.

Lastly, based on these similarities, we assign d the class label corresponding to the most similar centroid.

3. PROPOSED METHOD

3.1 Rationale

Let's take a simple example (see Figure 1(A)). The old-domain examples are represented by two circles: negative example is denoted by grey and positive example is denoted by white; the new-domain examples are represented by two ellipses: negative example is denoted by grey and positive example is denoted by white. C_{ON} and C_{OP} are the centroids of negative and positive class in old domain respectively. Middle Line is the perpendicular bisector of the line between C_{ON} and C_{OP}. From another perspective, Middle Line serves as a decision hyper-plane that separate negative class and positive class.

As a result, according the Middle Line, we can observe that all examples of old domain can be correctly classified. However, this Middle Line does not work well in new domain: from the figure, these new-domain negative examples under the Middle Line will be misclassified into positive class. This is the mechanism why domain-transfer degrades the performance of old classifier.

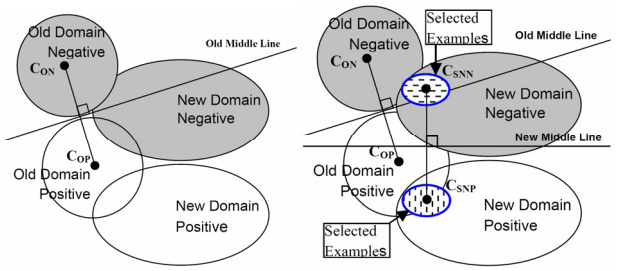

Figure 1: Performance of old classifier when transferred to new domain

An intuitive method to address this issue is to pick out some informative examples for new-domain and to train the base classifier again (see Figure 1(B)). We use "-" to represent selected examples from new-domain negative class, and use "|" to denote selected ones from new-domain positive class. Then we calculate two centroids, i.e., C_{SNN} and C_{SNP}, for these two new classes. As such, New Middle Line can be drawn. In this time, we can observe that most examples in new domain can be correctly categorized. This is the rationale why proposed scheme can address domain-transfer problem.

Detailed algorithm for proposed scheme is presented in Figure 2. Obviously, the most important and difficult job in this scheme is to label some informative ones for a new domain, because the old classifier performs poorly in new domain. In the following subsection we attempt to solve this problem.

1 Load old-domain labeled data (OL), new-domain unlabeled data (NU), and parameter "Ratio";
2 Train base classifier using labeled data in old domain;
3 Label some informative unlabeled ones in new domain;
4 Learn a new classifier using these selected examples;
5 Classify examples in new domain using new classifier.

Figure 2: Outline of Proposed Scheme for Domain-transfer Problem

3.2 Analysis and Method

Assuming X_1 denotes the word space of old domain, X_2 denotes the word space of new domain, and $X=X_1 \cup X_2$ denotes the total word space. We make following presumptions:

(i): Words in X are independent each other;

(ii): $X_\cap = X_1 \cap X_2 \neq \Phi$;

(iii): $X_\cap$ in different domains accords with the same probability distribution.

According to Bayesian formula, with respect to one examples d in old domain, the Bayesian discriminant function is

$$p(c \mid d) = \frac{p(d \mid c)p(c)}{p(d)} = \frac{p(d_{X_1 \setminus X_\cap}, d_{X_\cap} \mid c)p(c)}{p(d)} = \frac{p(d_{X_1 \setminus X_\cap} \mid c)p(d_{X_\cap} \mid c)p(c)}{p(d)}$$

$$g_1(d) = \ln(p(d_{X_1 \setminus X_\cap} \mid c)) + \ln(p(d_{X_\cap} \mid c)) + D_1 \quad (3)$$

where c denotes the class label, i.e., positive or negative, and D_1 indicates a constant.

As such, with respect to one examples e in new domain, the Bayesian discriminant function is

$$p(c \mid e) = \frac{p(e_{X_2 \setminus X_\cap} \mid c)p(e_{X_\cap} \mid c)p(c)}{p(e)}$$

$$g_2(e) = \ln(p(e_{X_2 \setminus X_\cap} \mid c)) + \ln(p(e_{X_\cap} \mid c)) + D_2 \quad (4)$$

where D_2 indicates a constant.

Consequently, if we directly apply the discriminant rule trained on old domain to new domain, the Bayesian discriminant function is

$$g_1(e) = \ln(p(e_{X_\cap} \mid c)) + D_1 \quad (5)$$

Obviously, there is a noticeable difference between above discriminant function and new-domain discriminant function. In most cases, accordingly, direct application of old-domain discriminant function (3) to classify new-domain examples is unfeasible. This is the mechanism why old-domain-trained classifier often performs extremely badly in new domain (as illustrated in Figure 1(A)).

However, $X_\cap$ and $X_2 \setminus X_\cap$ are independent each other. According to formulas (4) and (5), for every example e on new domain, we can make a conclusion that the larger $g_1(e)$ is, the larger $g_2(e)$ is.

Assuming the data in $X_2 \setminus X_\cap$ is in accord with a kind of distribution. As a result, with respect to examples e_1, e_2 in new domain, we obtain,

$$g_1(e_1) > g_1(e_2) \Rightarrow p(g_2(e_1) > g_2(e_2)) > p(g_2(e_1) < g_2(e_2)) \quad (6)$$

Above formula indicates that, if $g_1(e_1) > g_1(e_2)$, then the probability of $g_2(e_1) > g_2(e_2)$ is bigger than the probability of $g_2(e_1) < g_2(e_2)$.

With respect to centroid classifier, we can calculate its positive similarity (S^P) and negative similarity (S^N) using formula (2). Under this scenario, formula (6) leads to a conclusion: for one example, the larger the S^N, the more likely it is drawn from negative class; the larger the S^P, the more likely it is taken from positive class. Based on this conclusion, we propose Similarity Ranking method (SR): we first rank S^N of all examples, and assign top $n/2$ largest examples as negative; then rank S^P, and label top $n/2$ largest ones as positive.

However, this method doesn't hold when the length difference among different reviews is very large, because it is often the case that the larger the length of one review, the larger the S^N or S^P. What's worse, when transfer the old classifier to another domain, even if the actual length of reviews is nearly the same, the word-space difference between old domain and new domain can make a large difference on S^N or S^P.

To tackle this problem, we normalize (or divide) the original similarity so that the adverse effect of length difference and word-space variation can be offset to a high degree. This is the basic idea of relative similarity. Formally, we define Negative Relative Similarity (S^{RN}) and Positive Relative Similarity (S^{RP}) as following,

$$S^{RP} = \frac{S^P}{\left(S^N + S^P\right)/2}. \quad (7)$$

$$S^{RN} = \frac{S^N}{\left(S^N + S^P\right)/2}. \quad (8)$$

Up to this point, we can make a refined **conclusion** that, for one example, the larger the S^{RN}, the more likely it is drawn from negative class; the larger the S^{RP}, the more likely it is taken from positive class. According to this supposition, we propose Relative Similarity Ranking method (RSR): we first rank S^{RN} of all examples, and assign top $n/2$ largest examples as negative; then rank S^{RP}, and label top $n/2$ largest ones as positive.

In conclusion, we present the detailed algorithm for Relative Similarity Ranking. In this figure, Sizeof(NU) indicates the cardinality of unlabeled examples set in new domain.

1 Calculate similarity using formula (2);

2 Calculate Relative Similarity using formulas (7-8);

3 Rank S^{RN} and S^{RP} respectively;

4 Label top n/2 examples as Negative in S^{RN} list, and label top n/2 as positive in S^{RP} list, where n=Ratio*Sizeof(NU).

Figure 3: The Outline of Relative Similarity Ranking Method

3.3 A Case Study

Given Computer review (Comp) as old domain and House review (Hou) as new domain. For the sake of being easy to explain, we take out six examples (d_{15}, d_{20}, d_{49}, d_{63}, d_{76}, d_{114}) randomly (refer to Table 1) from Hou. The former three examples (d_{15}, d_{20}, d_{49}) are drawn from negative class, and the latter three (d_{63}, d_{76}, d_{114}) are coming from positive class.

For each example, we use formula (2) to calculate negative similarity (S^N) and positive similarity (S^P). In this time, centroid decision rule doesn't work at all: it classifies all of the six examples into negative class.

In accordance with SR, we first rank S^N and label examples (d_{15}, d_{20}, d_{63}) as negative; then rank S^P and assign (d_{49}, d_{76}, d_{114}) as positive (assuming n=6). Obviously, d_{63} and d_{49} are misclassified. This observation indicates that SR suffers from word-space difference incurred by domain-transfer.

To overcome this shortcoming of SR, we proposed RSR method in preceding subsection. Let's turn to Table 1 again. First we calculate relative similarity for six examples using formulas (7-8), then rank S^{RN} and label examples (d_{15}, d_{20}, d_{49}) as negative; rank S^{RP} and label examples (d_{63}, d_{76}, d_{114}) as positive (assuming n=6). In this time, all examples are correctly classified.

Table 1: Relative similarities of six randomly selected examples

Similarity / Examples	Original Similarity		Relative Similarity	
	S^N	S^P	S^{RN}	S^{RP}
d15	**0.6467(1)**	0.5844(1)	**1.0505**	0.9493
d20	**0.6178(3)**	0.5549(3)	**1.0537**	0.9464
d49	0.4252(5)	**0.3832(5)**	**1.0520**	0.9480
d63	**0.6206(2)**	0.5815(2)	1.0324	**0.9674**
d76	0.5939(4)	**0.5453(4)**	1.0427	**0.9573**
d114	0.2888(6)	**0.2622(6)**	1.0483	**0.9517**

4. EXPERIMENT RESULTS

4.1 Datasets

To validate the effectiveness and robustness of proposed method, we collected three datasets from three different domains: Computer Reviews (Comp), Education Reviews (Edu) and House Reviews (Hou). The details are listed in Table 2.

Table 2: The comparison of three datasets

	negative	positive	Averagelength	Vocabulary
Comp	390	544	120	4725
Edu	1012	254	600	19150
Hou	445	555	300	12674

4.2 Experimental Design

We use 50% of one dataset as training set when it is used as old domain. There is no doubt that feature selection may remove some important features in new domain, so we don't delete any features when training the base classifier in old domain.

Joachims's SVM-light package can be used for TSVM classification. (http://svmlight.joachims.org/). We use a linear kernel and leave all parameters as default. We use 50% of old-domain data as labeled training examples and use 50% of new-domain data as unlabeled ones.

4.3 Comparison and Analysis

Under the proposed scheme, we use two methods to pick out new-domain informative examples: Similarity Ranking (SR), and Relative Similarity Ranking (RSR). Both SR and RSR pick out some informative examples rather than label all examples in new

domain. We split the new-domain data evenly into unlabeled set and test set; the Ratio is set to 0.4 for SR and RSR.

As we can observe from table 3, RSR dramatically improves the performance of base classifier in new domain. The wide margin improvement indicates that proposed scheme combined with Relative Similarity Ranking method performs very effectively and robustly.

Despite of simplicity and straightforwardness, SR performs quite well. Its average accuracy is about 5% lower than RSR but about 18% higher than base classifier. For problems "Edu->Hou", SR even achieves better result than RSR. This performance provides convincing proof for the effectiveness of ranking based method.

In contrast to Centroid classifier, TSVM performs very well for domain-transfer problems. Apart from "Comp->Edu", TSVM beats Centroid classifier by a wide margin. The average accuracy of TSVM is about 12 percents higher than Centroid classifier. On other hand, however, TSVM is still outperformed by proposed scheme. For examples, apart from "Comp->Hou", TSVM is outperformed by SR or RSR scheme with wide margin. This observation indicates that proposed straightforward scheme can produce much better results than theoretically-more-sound TSVM.

Table 3: Accuracy when transferred to new domain

	Centroid	TSVM (Baseline)	Proposed Scheme	
			SR	RSR
Comp->Edu	0.7993	0.6887	0.6966	**0.8530**
Comp->Hou	0.4540	**0.8960**	0.8320	**0.8440**
Edu->Comp	0.5053	0.6509	0.7751	0.8051
Edu->Hou	0.5120	0.6100	**0.8280**	0.7200
Hou->Comp	0.7387	0.7815	0.8094	**0.8993**
Hou->Edu	0.5781	0.6840	0.7109	**0.8214**
Average	0.5979	0.7185	0.7753	**0.8238**

Figure 4 shows the performance curves of proposed method vs. the Ratio. It is worth noticing that we only use RSR to pick out informative examples. The parameter "Ratio" indicates what percentage of new-domain unlabeled data shall be picked out as informative examples. We split the new-domain data evenly into unlabeled set and test set.

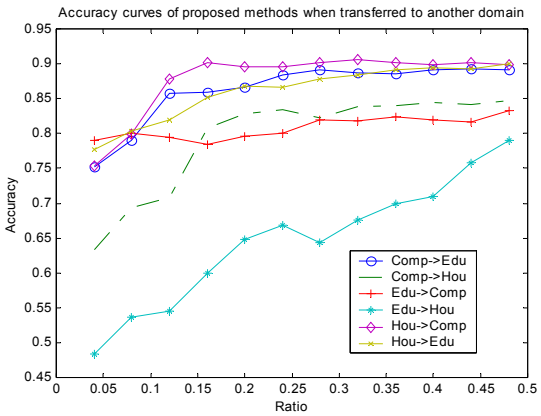

Figure 4: Accuracy curves of proposed method vs. the Ratio

We can clearly observe that increasing the Ratio increases the classification accuracy in new domain. However, the increase in accuracy is not directly proportional to the increase in the Ratio. As the Ratio gets larger, the accuracies start leveling off as we can observe from this figure apart from the problem "Edu->Hou".

The second observation is that, when the Ratio exceeds 0.15, proposed method achieves persistent results except "Edu->Hou" problem. This fact validates that the relative similarity method can pick out informative examples in new domain.

5. DISCUSSION

There are a few limitations with this work. First, although RSR can pick out "informative" examples in new domain, we still cannot guarantee that the selected "informative" ones are representative to the new domain. The second problem is that the chosen old domain may be far different from the new one, which makes it difficult to select really "informative" examples. Thirdly, when the negative examples are severely overlapped with the positive ones, RSR may label one same example into negative class as well as into positive class.

6. CONCLUSION REMARKS

In this work, we proposed an effective scheme for domain-transfer problem. It works by using old classifier to label some informative unlabeled examples in new domain, and training the base classifier again. To effectively pick out informative examples, we proposed Relative Similarity Ranking method. The main idea is to counteract the affect of domain-transfer by altering the original similarities. An empirical evaluation conducted on three domains indicates that proposed method dramatically boost the accuracy of the base sentiment classifier on new domain.

7. ACKNOWLEDGMENTS

This work was mainly supported by special fund of Chinese Academy of Sciences, "Research on Opinion Mining of Web Text", under grant number 0704021000 and two projects, i.e., 2007CB311100 and 2007AA01Z441.

REFERENCES

[1] Aue, A. and Gamon, M. Customizing Sentiment Classifiers to New Domains: a Case Study. RANLP, 2005.

[2] Finn, A., and Kushmerick, N. Learning to classify documents according to genre. In IJCAI-03 Workshop on Computational Approaches to Style Analysis and Synthesis. 2003

[3] Han, E. and Karypis, G. Centroid-Based Document Classification Analysis & Experimental Result. PKDD 2000.

[4] Mullen, T. and Collier, N. Sentiment analysis using support vector machines with diverse information sources. EMNLP. 2004, 412-418

[5] Pang, B., Lee, L., and Vaithyanathan, S. Thumbs up? Sentiment classification using machine learning techniques. EMNLP, 2002.

[6] Rijsbergen, C. Information Retrieval. Butterworths, London, 1979.

[7] Jing Jiang and ChengXiang Zhai. Instance weighting for domain adaptation in NLP. ACL 2007.

[8] Jing Jiang and ChengXiang Zhai. Exploiting domain structure for named entity recognition. HLT-NAACL 2006.